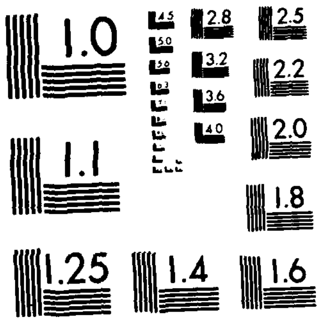
AD-A126 703 A MODEL OF THE TRAILING EDGE SEPARATION ON AN AIRFOIL 1/1
(U) VON KARMAN INST FOR FLUID DYNAMICS
RHODE-SAINT-GENESE (BELGIUM) G ZILLIAC FEB 83
UNCLASSIFIED EOARD-TR-83-7 AFOSR-82-0204 F/G 20/4 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

EOARD-TR-83-7

GRANT AFOSR 82-0204

AD A126703

A MODEL OF THE TRAILING EDGE SEPARATION ON AN AIRFOIL

GREG ZILLIAC
VON KARMAN INSTITUTE FOR FLUID DYNAMICS
CHAUSSÉE DE WATERLOO, 72
B - 1640 RHODE SAINT GENÈSE, BELGIUM

FEBRUARY 1983

FINAL SCIENTIFIC REPORT, 15 MAY 1982 - 14 FEBRUARY 1983

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

PREPARED FOR

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
BOLLING AF BASE, DC 20332

AND

EUROPEAN OFFICE OF AEROSPACE RESEARCH AND DEVELOPMENT
LONDON, UK.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER EOARD-TR-83-7	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A MODEL OF THE TRAILING EDGE SEPARATION ON AN AIRFOIL		5. TYPE OF REPORT & PERIOD COVERED Final Scient. Rep. 15 May 82 - 14 Feb 83
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Greg ZILLIAC		8. CONTRACT OR GRANT NUMBER(s) AFOSR 82-0204
9. PERFORMING ORGANIZATION NAME AND ADDRESS von Karman Institute for Fluid Dynamics, Chaussée de Waterloo, 72, B-1640 Rhode-Saint-Genèse, Belgium		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS P.E. 61102F Proj/Task 2301/D1
11. CONTROLLING OFFICE NAME AND ADDRESS European Office of Aerospace R&D/CA Box 14 FPO New York 09510		12. REPORT DATE February 1983
		13. NUMBER OF PAGES 91
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) European Office of Aerospace Research and Development/CA Box 14 FPO New York 09510		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release; distribution unlimited.

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

BOUNDARY LAYER SEPARATION; LIFT PREDICTION; TURBULENT BOUNDARY LAYER; TWO DIMENSIONAL FLOW; SEPARATED FLOW

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A model of the turbulent boundary layer separation of a two dimensional airfoil has been investigated. The model consists of distributing sources on the contour in the separated region of sufficient strength to maintain the constant pressure characteristic of trailing edge boundary layer separation. A model for the attached boundary layer has also been included. An accurate estimation of maximum lift, pitching moment and pressure distribution have been obtained. A correct drag prediction has not been accomplished.

EOARD-TR-83-7

This report has been reviewed by the EOARD Information Office and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Cary A. Fisher

CARY A. FISHER
Colonel, USAF
Chief Scientist

Jerry Ray Bettis

JERRY R. BETTIS
Lt Colonel, USAF
Deputy Commander

COPY INSPECTED 2

Availability Codes
Avail and/or
Dist Special
A
23 CP

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

There are many people who deserve recognition for their contributions to this work. The basic framework of the computer programs existed when I started to work on this project. These programs were written or modified mainly by S. Huo, and R.A. Van den Braembussche among others.

I would also like to give additional thanks to R.A. Van den Braembussche, who spend a lot of time helping me become familiar with the methods used.

Thanks also to Sharon Stanaway for her moral support and assistance in turning the lights off after long nights at the video terminal.

ABSTRACT

A model of the turbulent boundary layer separation of a two dimensional airfoil has been investigated. The model consists of distributing sources on the contour in the separated region of sufficient strength to maintain the constant pressure characteristic of trailing edge boundary layer separation. A model for the attached boundary layer has also been included. An accurate estimation of maximum lift, pitching moment and pressure distribution have been obtained. A correct drag prediction has not been accomplished.

LIST OF SYMBOLS

c_D	energy dissipation coefficient
c_f	skin friction coefficient
c	airfoil chord
f_x, f_y	singularity influence functions
m	exponent in Faulkner-Skan velocity profile
r	transformed boundary layer variable
S	distance along contour non dimensionalized by chord
S^*	distance along contour in separated region
u_e	local external velocity
u_τ	skin friction velocity
v	local velocity on airfoil in the separated region
w_L	local velocity on airfoil
z	point on airfoil where velocity is calculated
$B(,)$	boundary layer function
$C'(,)$	boundary layer function
C_d	two dimensional drag coefficient
C_ℓ	two dimensional lift coefficient
C_m	two dimensional pitching moment coefficient
C_p	pressure coefficient
$F( )$	Faulkner-Skan velocity profile function
G	function used in boundary layer analyses
H_{12}	momentum thickness form factor = δ^*/θ
H_{32}	energy thickness form factor = θ_H/θ
K	singularity influence matrix
L	Le Foll's form factor
p	pressure
Re_θ	Reynolds number based on the momentum thickness

TEPS,TESS	points just before airfoil trailing edge
U	component of the complex velocity in x-direction
V	component of the complex velocity in y-direction
W	complex velocity
X	Le Foll's Reynolds number
α	airfoil angle of attack or uniform stream basic flow vortex strength
β	uniform stream basic flow vortex strength
γ	total vortex strength
δ	outflow basic flow vortex strength
δ^*	boundary layer displacement thickness
ζ	point where singularity is located
η	Blasius variable
θ	boundary layer momentum thickness
θ_H	boundary layer energy thickness
λ	Le Foll's parameter for the velocity profile
μ	circulatory basic flow vortex strength
ν	kinematic viscosity
ξ,η	coordinate of the singularity point location
ρ_∞	freestream density
τ	shearing stress
ϕ	airfoil transformed coordinates
Γ	total circulation
π	pressure gradient

LIST OF FIGURES

1. INTRODUCTION

Analytical solutions to the problem of boundary layer separation have long been sought after but few have been success ful. The Navier-Stokes equations which govern flow separation have not been solved for high Reynolds number flows, hence existing analytical solutions must involve assumptions concerning the nature of separation. Empericism and/or experimental results are often used. In these cases, the question arises as to whether we are obtaining solutions which accurately reflect the physics of the problem, or are we modeling a phenomenon. This paper deals with the latter of these possibilities.

One of the first and arguably the best analytical model of turbulent boundary layer separation was developed Jacob (Ref. 1). This model consists of using inviscid singularity methods to model a viscous phenomenon. Sources are distributed in the dead air region to maintain constant pressure (in this region). A constant pressure, dead air region is a characteristic of turbulent boundary layer separation on an airfoil. The boundary layer and outer inviscid region must simultaneously be accounted for to complete the picture. Jacob has used this model to achieve very accurate prediction of maximum lift.

A second model which has been used with similar success is the model proposed by Dvorak and Maskew (Ref. 2). Their model consists of placing vortices on the free shear layer. The strength of the vortices is such as to obtain constant pressure in the separated region.

There are many other methods which have a stronger physical basis, but surprisingly, do not give results which can compare with the quality of results of Jacob and Dvorak's techniques.

In this investigation, Jacob's model has been chosen, because it has been in existance longer and has proven its validity as a model.

It is the object of this investigation to develop the method to the point where an accurate estimation of the lift, drag, pitching moment and pressure distribution can be obtained for the turbulent boundary layer separation of a two dimensional airfoil.

The importance of this investigation is that a correct prediction of the maximum lift coefficient is required. An inaccurate estimation of $C_{\ell_{max}}$ can have drastic effects on the payload capacity of an aircraft (under estimation of $C_{\ell_{max}}$ by 0.1 can mean up to a 20% reduction in payload).

Two points must be emphasized before continuing. First, this method is only applicable as a design tool and will not give any insight into the mechanisms of separation. Secondly, separation is a three dimensional, and quite often unsteady phenomenon. This has been demonstrated by wind tunnel tests which have shown that under optimum conditions, it is difficult to obtain a truly two dimensional and steady measurement. Hence, the results of this research cannot directly be extended to a three dimensional situation.

2. THEORETICAL FOUNDATION OF THE MODEL

2.1 The model

The model is shown in figure 1. It consists of breaking down the separated flow over an airfoil into three regions :

Region 1 : the outer inviscid region which is modelled by line vortices placed on the contour of the airfoil;

Region 2 : the boundary layer characteristics which are determined by an integral boundary method (similar to Le Foll's method). The boundary layer effect is included by distributing sources to compensate for the displacement thickness.

Region 3 : the separated region is modelled by sources distributed on the contour of sufficient strength to maintain constant pressure on the airfoil in this region.

2.2 The inviscid and separated regimes

2.2.1 Governing equations

The governing equation for the model can be derived from the boundary contours. Three boundary conditions are required for the analysis. Application of these conditions gives a Glauert type integral expression for the velocity on the contour :

(1) Uniform stream far upstream.

(2) Normal velocity on the airfoil surface is zero (kinematic condition).

(3) An equivalent Kutta condition.

In addition to the boundary conditions, an additional criterium that the pressure is constant in the separated region must be enforced. Conditions 1 & 2 give rise to the integral expression for the vortex strength in function of blade geometry, uniform velocity, and the source distribution.

$$U_\infty \frac{dy}{dS} - V_\infty \frac{dx}{dS} = \frac{1}{2\pi} \oint \gamma(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta)$$

$$+ \frac{1}{2\pi} \oint Q(\zeta) \left[-f_x(z,\zeta) \frac{dx}{dS} + f_y(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) \qquad (2.1)$$

where:

$$f_y(z,\zeta) = \frac{(x-\xi)}{(x-\xi)^2+(y-\eta)^2}$$

$$f_x(z,\zeta) = \frac{-(y-\eta)}{(x-\xi)^2+(y-\eta)^2}$$

Martensen (Ref. 3) has shown that the flow over an airfoil may be considered as the superposition of four basic flows, a uniform stream in the x-direction, a uniform stream in the y-direction, a circulatory flow, and an outflow. Expression (2.1) becomes four integral expressions for the vortex strength of the basic flows.

Uniform stream in x-direction :

$$\frac{1}{2\pi} \oint \alpha(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) = \frac{dy}{dS} \qquad (2.2)$$

Uniform stream in y-direction :

$$\frac{1}{2\pi} \oint \beta(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dx}{dS} \right] dS(\zeta) = \frac{dx}{dS} \qquad (2.3)$$

Circulatory flow :

$$\frac{1}{2} \oint \mu(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) = 0 \qquad (2.4)$$

Outflow :

$$\frac{1}{2\pi} \oint \delta(\theta) \left[f_y(z,\zeta) \frac{dx}{DS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta)$$

$$= - \frac{1}{2\pi} \oint Q(\zeta) \left[f_x(z,\zeta) \frac{dx}{dS} - f_y(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) \qquad (2.5)$$

Equation (2.1) has been split into one homogeneous and three non-homogeneous equations. The total solution is given by :

$$\gamma(\zeta) = U_\infty \gamma(\zeta) - V_\infty \beta(\zeta) + \Gamma \mu(\zeta) + \delta(\zeta) \qquad (2.6)$$

The following supplementary equations are required because the circulatory basic flow is the only basic flow which contributes to the circulation (note Γ in equation 2.6) :

$$\oint \zeta(\zeta) dS(\zeta) = 0 \qquad (2.7)$$

$$\oint \beta(\zeta) dS(\zeta) = 0 \qquad (2.8)$$

$$\oint \mu(\zeta) dS(\zeta) = 1 \qquad (2.9)$$

$$\oint \delta(\zeta) dS(\zeta) = 0 \qquad (2.10)$$

Applying the equivalent Kutta condition determines the magnitude of the total circulation :

$$\gamma(TEPS) = - \gamma(TESS) \qquad (2.11)$$

Solving for the circulation :

$$\Gamma = - \frac{U_\infty[\alpha(TEPS)+\alpha(TESS)] + V_\infty[\beta(TEPS)+\beta(TESS)] + [\delta(TEPS)+\delta(TESS)]}{[\mu(TEPS)+\mu(TESS)]}$$

The local velocity on the contour is given by :

$$W_L = \sqrt{\gamma(\zeta)^2 + Q(\zeta)^2} \tag{2.13}$$

This expression contains a contribution due to the vortices and due to the sources (sources are distributed along the contour for the boundary layer model as well as the separation model).

It appears that the complexity of the problem has been increased by separating the flow into four basic flows and that is true. The major advantage of this approach is that the velocity on the contour for any angle of attack can be determined by substitution of the corresponding freestream velocity components in equation (2.6). One limitation is that if the source distributions is changed, the whole problem must be resolved.

The solution of these integral equations is not straightforward. The equations become singular when calculating the influence of a particular vortex (or source) at its origin. This difficulty can be avoided as shown in appendix A.

2.2.2 Numerical considerations

Attention must now be turned to the numerical side of the problem. The profile is placed in the x-y plane as shown in figure 2. A coordinate transformation is used :

$$x = \frac{c}{2}(1-\cos\phi) \tag{2.14}$$

This transformation allows a greater number of singularities to be placed near the leading and trailing edges where the slope of the contour changes rapidly. The contour is divided into N segments of constant included angle $\Delta\phi$. On each segment, a line vortex of constant strength is placed. Line vortices have an advantage over point vortices in that a line vortex distribution more closely approximates the continuous vortex

distribution of the integral equations. The integral expressions are replaced by the following summations :

Uniform stream in x-direction :

$$\frac{1}{2\pi} \sum_{i=1}^{N} \alpha(\zeta_i) K(z_j, \zeta_i, \Delta\zeta_i) = \frac{dy_j}{dS} \tag{2.15}$$

Uniform stream in y-direction :

$$\frac{1}{2\pi} \sum_{i=1}^{N} \beta(\zeta_i) K(z_j . \zeta_i, \Delta\zeta_i) = \frac{dx_j}{dS} \tag{2.16}$$

Circulatory flow :

$$\frac{1}{2\pi} \sum_{i=1}^{N} \mu(\zeta_i) K(z_j, \zeta_i, \Delta\zeta_i) = 0 \tag{2.17}$$

Outflow :

$$\frac{1}{2\pi} \sum_{i=1}^{N} \delta(\zeta_i) K(z_j, \zeta_i, \Delta\zeta_i) = \frac{1}{2\pi} \sum_{a}^{b} Q(\zeta_i) K(z_j, \zeta_i, \Delta\zeta_i) \tag{2.18}$$

where the influence function K is given in appendix A.

The supplementary conditions become :

$$\sum_{i=1}^{N} \alpha(\zeta_i) \Delta S(\zeta_i) = 0 \tag{2.20}$$

$$\sum_{i=1}^{N} \beta(\zeta_i) \Delta S(\zeta_i) = 0 \tag{2.21}$$

$$\sum_{i=1}^{N} \mu(\zeta_i) \Delta S(\zeta_i) = 1 \tag{2.22}$$

$$\sum_{i=1}^{N} \delta(\zeta_i) \Delta S(\zeta_i) = 0 \tag{2.23}$$

Supposing that equations (2.15) through (2.19) must be satisfied in the points j=1,N where the vortices are located and together with equations (2.20) through (2.23) one obtains a set of 4(N+1) equations with 4N unknowns. The unknowns are the vortex strength of the basic flows. However, a system of equations derived from an integral equation of the first kind is often ill-conditioned. This can be avoided if the kinetic condition is also applied at the intersection points of the line vortex. This leads to a set of 4(2N+1) equations with 4N unknowns, which can be treated by a least square technique. See Appendix A for further details.

2.3 The boundary layer analysis

2.3.1 The integral boundary layer approach

The method used for the boundary layer region is by S. Huo (Ref. 4) and is similar to Le Foll's method. Huo's method is based on two assumptions :

(1) Local boundary layer velocity profile similarity.

(2) The velocity profile is assumed to be a one parameter profile.

The integral boundary layer equations are :

Momentum :

$$d(\theta u_e^2) + \delta^* u_e du_e = c_f \frac{u_e^2}{2} dS \qquad (2.24)$$

Energy :

$$d\left(\theta_H \frac{u_e^3}{2}\right) = c_D u_e^3 dS \qquad (2.25)$$

where the skin friction coefficient is :

$$c_f = \frac{2\tau_w}{\rho u_e^2} \tag{2.26}$$

and the energy dissipation coefficient is :

$$c_D = \frac{2}{\rho u_\infty^3} \int_0^\infty \tau(y) \frac{\partial u}{\partial y} dy \tag{2.27}$$

The unknowns in equations (2.24) and (2.25) are $\delta^*, \theta, \theta_H, c_f$ and c_D. There are two equations at hand with five unknowns. In order to solve these equations, the following information is necessary :

(1) A velocity profile law in the boundary layer.

(2) A skin friction law.

(3) A relation for the dissipation coefficient.

The velocity profile law will enable the determination of the relation between the shape factors H_{12} and H_{32}. Therefore, knowing two of the characteristic thicknesses of the boundary layer from equations (2.24) and (2.25) will allow the calculation of the other thicknesses. The skin friction and dissipation coefficients will be determined from the skin friction law and dissipation relation respectively. Hence, the complete boundary layer characteristics can be computed.

Le Foll bases his method on the idea that the boundary layer can be described in terms of a form factor and a characteristic length instead of the velocity and length. The main advantage of this approach is that the inverse procedure of specifying boundary layer characteristics and their solving for the corresponding velocity distribution is easily achieved. The inverse problem is not of interest in this investigation, but the method is rapidly applicable to the direct problem.

The parameters L (Truckenbrodt's parameter) and X are used by Le Foll to specify the boundary layer characteristics:

$$dL = \frac{1}{(H_{12}-1)} \frac{dH_{32}}{H_{32}} \qquad X = \ln Re_{\theta_H} + 2L \qquad (2.28)$$

As they are directly related to each other, L and X can, therefore, replace H_{12} and Re_θ to characterize a boundary layer.

2.3.2 The laminar boundary layer

The assumed velocity profile is the Faulkner-Skan profile :

$$\frac{u}{u_e} = F(n,m) \qquad n = y\sqrt{\frac{u_e}{S}} \qquad u_e = S^m \qquad (2.29)$$

Stewartson's transformation has been used with the Faulkner-Skan profile to extend the boundary layer calculation into the compressible regime.

The compressibility aspect of boundary layer separation is not included in the separation model, but it was felt that since a compressible boundary layer method was available and programmed), compressibility should be (and was) included in the boundary layer model.

The skin friction and dissipation relation are functions of the assumed velocity profile. For the laminar boundary layer :

$$c_f = -\frac{4a_2(m)}{\sqrt{Re_x}} \qquad c_D = \frac{F_1(m)}{\sqrt{Re_x}} \qquad (2.30)$$

To simplify the calculation, another change of variable is introduced :

$$r = \frac{Re_{\theta H} e^{2L}}{Re_{ref}} \tag{2.31}$$

The momentum and energy integral equations become :

$$dr = u_e C(L,r) dS = \frac{u_e C'(L)}{r Re_{ref}} dS \tag{2.32}$$

$$\frac{dL}{dS} = \frac{1}{u_e} \frac{du_e}{dS} - B(L,r) \frac{1}{r} \frac{dr}{dS} \tag{2.33}$$

The function C'(L,r) and B(L,r) are given in reference 5.

Equations (2.32) and (2.33) are two first order differential equations which are solved by a fourth order Runge-Kutta procedure.

2.3.3 Transition and separation

Transition from laminar to turbulent flow can occur if the boundary layer becomes unstable (dictated by Schlichting's stability curve). The unstable region is :

$$L + 0.011 - 0.005\ X < 0 \tag{2.34}$$

Whether transition will actually occur depends on the turbulence level, pressure gradient, and Re_θ. A transition criterium has been developed by B. Roberts (Ref. 6) based on the experimental results of P.S. Granville. This criteria predicts transition if ΔRe_θ (the difference in Reynolds number between point in instability and point of transition) is larger than a specified amount (see Ref. 6).

The boundary layer may separate (laminar separation) before transition. Zero wall shear stress is used as the separation criteria. In the calculation of L, the integration constant is chosen so that L equals zero at separation. This corresponds to :

$H_{12} = 4.03 \qquad\qquad H_{32} = 1.515$

Since the inviscid model makes no allowences for laminar separation regions, some approximations must be introduced. In all cases laminar separation bubbles followed by turbulent reattachment is suspected, the length of the bubble is neglected and a turbulent boundary layer is assumed from the point of laminar separation.

2.3.4 The turbulent boundary layer

The assumed velocity profile is a modified Falkner-Skan profile :

$$\frac{u}{u_e} = 1 - \frac{u_\tau}{u_e}\frac{g(\eta,2)}{K} \qquad (2.35)$$

The function $g(\eta,Z)$ in equation (2.35) is determined from a combination of experimental results and theory (see Ref. 5 for further details). The turbulent boundary layer is also applicable to compressible flow by the inclusion of Van Driest's generalized velocity.

The skin friction coefficient is determined from :

$$\frac{u_\tau}{u_e} = \frac{K}{\lambda-.5-\ell n\left(\frac{2/3+\lambda/2}{K}\right)+KC+\ell n Re_{\delta^*}} \qquad (2.36)$$

The momentum and energy integral equations may be combined to obtain :

$$c_D = f(\delta ,\theta,\theta_H)c_f \tag{2.37}$$

If or a given Mach number, similar solutions exist, a relation between the pressure gradient and velocity profile parameter G must also exist. Assume that :

$$\pi = \left(\frac{G+1.635}{6.1}\right)^2 - 1.8 \tag{2.38}$$

where :

$$\pi = \frac{\delta_i^*}{\tau_w}\frac{dp}{dS}$$

and :

$$G = \frac{u_e}{u_\tau}\left(1 - \frac{1}{H_{12_i}}\right)$$

with these assumptions, sufficient information is available to solve the turbulent boundary layer characteristics. See reference 5 for further details.

Turbulent boundary layer separation is also indicated when L equals 0. At this point, the boundary layer calculation is stopped.

2.4 The source distribution

A source distribution is required for the separation region model. This distribution must be chosen in such a way as to maintain constant pressure (constant resultant velocity) in the separation region. Equation (2.1) which is in effect an expression for the resultant velocity on the airfoil indicates

that the source distribution should be a function of :

(1) profile geometry;

(2) upstream condition;

(3) angle of attack;

(4) separation point location;

(5) velocity distribution.

This is a paradoxical situation. The source distribution depends on the velocity on the contour which in turn depends on the source distribution. Also, for a given airfoil, the fundamental relationship for the source distribution must be general enough to allow for changes in angle of attack. To change the source function for different angles of attack would defeat the purpose of using a functional relationship.

Figure 3 shows a typical source distribution used in this study. The functional relationship is :

$$Q = 2.5S^{*}V_{sep.}\sin\alpha \qquad 0 \leqslant S^{*} \leqslant 0.4$$

$$Q = V_{sep.}\sin\alpha$$

$$Q = V_{sep.}\sin\alpha-(V_{sep.}\sin\alpha-0.2)(5S^{*}-3)^{2} / 4$$

where :

S^{*} is the distance along the contour in the separated region

$V_{sep.}$ is the velocity in the separated region

This relationship is dependent on angle of attack, velocity in the separated point and location of separation point. It does not include influence of the velocity distribution (over the remainder of the airfoil) on the separated region or influence of the profile geometry. A function which includes all of these aspects was too difficult to determine.

A completely different approach to the problem, which avoids functional source distributions altogether was attempted. This involves an iterative procedure to find a source distribution maintained constant velocity in the separated region, but this procedure failed to converge.

Another possible approach would be to a priori specify the velocity in the separated region, and then analytically solve the inverse problem to find the source strength. There have been recent research efforts at VKI to model wing tunnel wall interference effects using source distributions determined by an inverse procedure. This research has shown that a solution of a nearly singular system of equations is necessary to find the appropriate source distribution. This sheds a little light on the question why the iterative processes failed.

In the unseparated region a source distribution for the boundary layer model is determined from :

$$Q = \frac{d}{dS}\,(u_e \delta^*) \tag{2.40}$$

This relationship determines the source strength required to compensate for the displacement thickness. This approach (the blowing approach) is considered easier to use than the alternative approach of using a displacement surface for a couple of reasons : the profile geometry does not have to be recalculated. The blowing approach is also more compatible with the separated flow model.

2.5 Forces and moment on the airfoil

2.5.1 The pressure coefficient

The pressure coefficient is computed from the classical relationship :

$$c_p = 1 - \frac{w_L^2}{u_\infty^2} \tag{2.41}$$

where :

$$c_p = \frac{p-p_\infty}{q_\infty} \qquad q_\infty = \frac{1}{2}\rho u_\infty^2 \tag{2.42}$$

These expressions are a result of Bernoulli's equation and are valid because this is an inviscid model. It is assumed that the pressure on the airfoil is the same as the pressure along the free streamline. Also, no wake effects have been included.

2.5.2 Lift coefficient

The lift coefficient can be computed directly from the Kutta Joukowski law :

$$\Gamma = \oint \gamma dS$$

$$L = \Gamma \rho u_\infty \tag{2.43}$$

$$c_\ell = 2\Gamma$$

2.5.3 Drag coefficient

An accurate determination of the drag for an airfoil with turbulent boundary layer separation is very difficult due to the following :

(1) Integral boundary layer methods often do not predict the separation point exactly because these methods are not valid close to the separation points.

(2) Singularity methods are not accurate near the airfoil trailing edge.

(3) Surface roughness and freestream turbulence have a large influence.

(4) Wake effects have an effect especially when there is separation.

(5) The equivalent Kutta condition does not model the physical situation exactly.

(6) Deviations from constant pressure in the separated region can have a significant effect.

With these considerations in mind, the total drag is composed of a pressure contribution due to incomplete pressure recovery, and a viscous contribution due to the viscous shearing stresses. The viscous drag is :

$$C_{d_v} = \frac{D_v}{\frac{1}{2}\rho u_\infty^2 c} = \cos\alpha \int_0^{2\pi} c_f(\phi)\,\frac{dx}{d\phi}\left(\frac{w_L}{u_\infty}\right)^2 d\phi$$

$$+ \sin\alpha \int_0^{2\pi} c_f(\phi)\,\frac{dy}{d\phi}\left(\frac{w_L}{u_\infty}\right)^2 d\phi \qquad (2.44)$$

The pressure drag is :

$$c_{d_p} = \frac{D_p}{\frac{1}{2}\rho u_\infty^2 c} = -\cos\alpha \int_0^{2\pi} C_p(\phi)\,\frac{dy}{d\phi}\,d\phi$$

$$+ \sin\alpha \int^{2\pi} C_p(\phi)\,\frac{dx}{d\phi}\,d\phi \qquad (2.45)$$

The contribution of the separated region to the viscous drag is zero (c_f is zero in this region).

2.5.4 Pitching moment coefficient

The pitching moment is the integral of the pressure distribution over the contour and is given by :

$$C_{m_{1/4c}} = \frac{M}{\frac{1}{2}\rho u_\infty^2 c^2} = -\int_0^{2\pi} C_p(0.25-X)\,\frac{dx}{d\phi}\,d\phi$$

$$+\int_0^{2\pi} C_p(0.25-X)\,\frac{dx}{d\phi}\,d\phi \qquad (2.46)$$

Expression (2.46) is the pitching moment coefficient about the quarter chord.

2.6 The computer program

Four computer programs have been written which perform the calculations described in the preceeding sections. The basic computational procedure is outlined in figure 4 and listings of the program along with examples of the input and output are in Appendix B.

The calculations have been split into four programs to facilitate ease of use and also to allow modification of the results of each program if required. Duplication of computation is also avoided, for example, if for a given airfoil, the angle of attack is changed, the calculation may start with the second program (instead of the first program).

3. RESULTS

The separated turbulent boundary layer for the GA(W-1) and the NACA 2412 airfoil have been investigated. The results of these invesitgations are presented in figures 5 through 18.

For the GA(W-1) airfoil, figures 5 and 6 show the inviscid and viscid pressure coefficient for low angles of attack. Notice that the viscous calculated pressure coefficient agrees very well with the experimental results (Ref. 7) except near the leading edge where there is a short laminar separation bubble.

At angles of attack greater than four degrees, computational difficulties arose. The boundary layer program was originally written and calibrated for turbomachinery applications and in cases where the pressure peaks near the leading edge become too great, the boundary layer calculation failed. A second problem was a short laminar separation bubble which existed throughout the angle of attack range.

Figures 7-14 show the results of the calculation where the experimental separation point has been used. The boundary layer model has not been included. The source distribution used for the separation region has been optimized for an angle of attack of about 18 degrees, hence for other angles of attack, the pressure distribution is not as constant (in the separation region) as it is for 18 degrees.

Figures 15 and 16 show the lift and pitching moment coefficient versus angle of attack respectively. Agreement with the experimental results is quite good. There is a deviation of the computed pitching moment coefficient at the high angles of attack. This is due to the calculated pressure distribution which differs slightly from the experimental pressure distribution for the high angle of attack cases. Since the boundary layer program failed, a calculation of the drag was not achieved.

Figure 17 shows the results of a computed pressure distribution for NACA 2414 airfoil. A very flat pressure distribution was obtained for this angle of attack. The lift curve for this airfoil is presented in figure 18. To obtain results for the NACA 2412 which included a prediction of the separation point the computational procedure was again modified.

(1) The extremely high pressure peak near the leading edge was lowered (first five percent of airfoil).

(2) An estimated separation point was used.

(3) The boundary layer program was then used to validate the estimated separation point location.

It was felt that lowering the pressure peak was justifiable because the boundary layer program predicted laminar separation in the first five per cent. A short laminar separation bubble would lower the pressure peak and since there is no laminar separation model, the peak was lowered by using engineering judgement. Figure 17 shows that a good prediction of $C_{\ell_{max}}$ was obtained.

4. CONCLUSIONS AND RECOMMENDATIONS

The following can be concluded about the investigation :

(1) The model appears to be a valid model for turbulent boundary layer separation.

(2) Could improve the method of determining the source distribution for the separated region. The inverse problem is a possibility. It will be very difficult to define an iterative procedure to find the source strength because of convergence problems.

(3) The boundary layer program should be extended so that it can handle airfoils with high pressure peaks and at high Reynolds number.

(4) A laminar separation model should be included.

(5) The lift, pitching moment, and pressure distribution have been accurately predicted for two airfoils.

(6) The drag has not been predicted.

(7) A wake model should be incorporated.

(8) Extension of the computer model to compressible flow is worth considering.

REFERENCES

1. JACOB, K.: Berechnung der abgelösten inkompressiblen Strömung um Tragflügelprofile und Bestimmung des maximalen Auftriebs.
Zeitschrift für Flugwissenschaften,
Bd. 17, Heft 7, Juli 1969, pp 221-230.

2. DVORAK, F.A. & MASKEW, B.: Application of the AMI $C_{\ell max}$ prediction method to a number of airfoils. In: Advanced Technology Airfoil Research.
NASA CP 2045, Vol. 1, 1978.

3. MARTENSEN, E.: Berechnung der Druckverteilung an Gitter-Profilen in ebener Potentialströmung mit einer Fredholmschen Integralgleichung.
Archive for Rational Mech. and Analysis,
Vol. 3, No. 3, 1959, pp 235.

4. HUO, S.: Optimization based on the boundary layer concept for compressible decelerating flows.
Ph.D. Thesis, ULB, October 1973.

5. PAPAILIOU, K.: Blade optimization based on boundary layer concepts.
VKI CN 60, 1967.

6. ROBERTS, W.B.: A study of the effects of Reynolds number and laminar separation bubbles on the flow through axial compressor cascades.
Ph.D. Thesis, ULB, May 1973.

7. McGHEE, R.J. & BEASLEY, W.D.: Low speed aerodynamic characteristics of a 17-percent-thick airfoil section designed for general aviation applications.
NASA TN D-7428, 1973.

APPENDIX A - DERIVATION AND SOLUTION OF THE INTEGRAL EQUATIONS

Derivation of the integral equations

This appendix contains a detailed derivation of integral equations. Attention is also given to the solution of these equations.

The complex velocity at point x,y induced by a vortex at ξ,η is :

$$\bar{W}(z) = u - iv = -\frac{i}{2\pi}\frac{\gamma(\rho)}{X+iY} \tag{A.1}$$

where :

$X = x-\xi$ and $Y = y-\eta$

The complex velocity due to a distribution of vortices on a closed contour is :

$$\bar{W}(z) = -\frac{i}{2\pi}\oint\frac{\gamma(\zeta)}{X+iY}\,dS(\zeta) \tag{A.2}$$

separating the complex velocity into its components :

$$U(z) = -\frac{1}{2\pi}\oint\frac{Y\gamma(\delta)}{X^2+Y^2}\,dS(\zeta) \tag{A.3}$$

$$V(z) = \frac{1}{2\pi}\oint\frac{X\gamma(\rho)}{X^2+Y^2} \tag{A.4}$$

A uniform stream must be superimposed :

$$\bar{W}_\infty = U_\infty + iV_\infty \tag{A.4}$$

$$U(z) = U_\infty - \frac{1}{2\pi} \oint \frac{Y\gamma(\zeta)}{X^2+Y^2}\, dS(\zeta) \qquad (A.6)$$

$$V(z) = V_\infty + \frac{1}{2\pi} \oint \frac{X\gamma(\zeta)}{X^2+Y^2}\, dS(\zeta) \qquad (A.7)$$

Sources are placed on the contour for the boundary layer model.

$$W(z) = \frac{1}{2\pi} \frac{Q(\zeta)}{X+iY} \qquad (A.8)$$

For sources distributed on a contour, the complex velocity is :

$$W(z) = \frac{1}{2\pi} \oint \frac{Q(\zeta)}{X+iY}\, dS(\zeta) \qquad (A.9)$$

Superimposing the source distribution :

$$U(z) = U_\infty - \frac{1}{2\pi} \oint \frac{Y\gamma(\zeta)}{X^2+Y^2}\, dS(\zeta) + \frac{1}{2\pi} \oint \frac{XQ(\zeta)}{X^2+Y^2}\, dS(\zeta) \qquad (A.10)$$

$$V(z) = V_\infty + \frac{1}{2\pi} \quad \frac{\gamma\ \zeta}{X^2+Y^2}\, dS(\zeta) - \frac{1}{2\pi} \oint \frac{YQ(\zeta)}{X^2+Y^2}\, dS(\zeta) \qquad (A.11)$$

The kinetic boundary condition that the velocity normal to the airfoil surface is zero is enforced. This is equivalent to stating that the surface is a streamline.

$$\frac{\partial \psi}{\partial S} = 0$$

$$\frac{\partial\psi}{\partial S} = \frac{\partial\psi}{\partial x}\frac{dx}{dS} + \frac{\partial\psi}{\partial y}\frac{dy}{dS}$$

$$0 = U(z)\frac{dy}{dS} - V(z)\frac{dx}{dS} \tag{A.12}$$

Introducing (A.10) and (A.11) one obtains :

$$\begin{aligned} 0 = U_\infty \frac{dy}{dS} &- \left[\frac{1}{2\pi}\oint \frac{Y\gamma(\zeta)}{X^2+Y^2}\,dS(\zeta)\right]\frac{dy}{dS} \\ &+ \left[\frac{1}{2\pi}\oint \frac{XQ(\zeta)}{X^2+Y^2}\,dS(\zeta)\right]\frac{dy}{dS} \\ - V_\infty \frac{dx}{dS} &- \left[\frac{1}{2\pi}\oint \frac{X\gamma(\zeta)}{X^2+Y^2}\,dS(\zeta)\right]\frac{dx}{dS} \\ &+ \left[\frac{1}{2\pi}\oint \frac{YQ(\zeta)}{X^2+Y^2}\,dS(\zeta)\right]\frac{dx}{dS} \end{aligned} \tag{A.13}$$

Since $dS(\zeta)$ is not a function of x,y the derivatives $\frac{dy}{dS}$ and $\frac{dx}{dS}$ can be taken inside the integral.

$$\begin{aligned} U_\infty \frac{dy}{dS} - V_\infty \frac{dx}{dS} &= \frac{1}{2\pi}\oint \gamma(\zeta)\left[f_y(z,\zeta)\frac{dx}{dS} - f_x(z,\zeta)\frac{dy}{dS}\right] dS(\zeta) \\ &+ \frac{1}{2\pi}\oint Q(\zeta)\left[f_x(z,\zeta)\frac{dx}{dS} - f_y(z,\zeta)\frac{dy}{dS}\right] dS(\zeta) \end{aligned} \tag{A.14}$$

where :

$$f_y(z,\zeta) = \frac{X}{X^2+Y^2} \qquad\qquad f_x(z,\zeta) = -\frac{Y}{X^2+Y^2}$$

Integral expression (A.14) can be split into one homogeneous and three non-homogeneous equations corresponding to the four basic flows described in chapter 2.

$$\frac{1}{2\pi} \oint \alpha(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) = \frac{dy}{dS} \tag{A.15}$$

$$\frac{1}{2\pi} \oint \beta(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) = \frac{dx}{dS} \tag{A.16}$$

$$\frac{1}{2\pi} \oint \mu(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS = 0 \tag{A.17}$$

$$\frac{1}{2\pi} \oint \delta(\zeta) \left[f_y(z,\zeta) \frac{dx}{dS} - f_x(z,\zeta) \frac{dy}{dS} \right] dS(\zeta) \tag{A.18}$$

$$= - \frac{1}{2\pi} \oint Q(\zeta) \left[f_x(z,\zeta) \frac{dx}{dS} - f_y(z,\zeta) \frac{dy}{dS} \right] dS(\zeta)$$

The total vortex strength is the superposition of the four basic vortex strengths.

$$\gamma(\zeta) = U_\infty \alpha(\zeta) - V_\infty \beta(\zeta) + \Gamma \mu(\zeta) + \delta(\zeta) \tag{A.19}$$

Equations (A.15) through (A.18) are obtained by substituting equation (A.19) into (A.14) and collecting like terms.

The far field boundary condition (uniform stream far upstream) has been implicitly satisfied. This can be demonstrated by taking the limit of (A.10) and (A.11). As x approaches infinity, the velocity components become equivalent to the imposed uniform stream. This result shows that the singularities do not have an influence far upstream.

The supplementary equations (as described in chapter 2) are :

$$\oint \alpha(\zeta)dS(\zeta) = 0 \qquad (A.20)$$

$$\oint \beta(\zeta)dS(\zeta) = 0 \qquad (A.21)$$

$$\oint \mu(\zeta)dS(\zeta) = 1 \qquad (A.22)$$

$$\oint \delta(\zeta)dS(\zeta) = 0 \qquad (A.23)$$

The equivalent Kutta condition fixes the circulation :

$$\gamma(TEPS) = - \gamma(TESS) \qquad (A.24)$$

Using equations (A.19 and A.24) gives :

$$\Gamma = \frac{-U_\infty[\alpha(TEPS)+\alpha(TESS)]+V_\infty[\beta(TEPS)+\beta(TESS)]+[\delta(TEPS)+\delta(TESS)]}{[\mu(TEPS)+\mu(TESS)]} \qquad (A.25)$$

Solution of the integral equations

The solution of the integral equations is complicated by the fact that expressions (A.15) through (A.18) are singular integrals. Before the discussion of the singular integrals, a coordinate transformation which allows greater numerical accuracy will be performed :

$$x = \frac{c}{2}(1-\cos\phi) \qquad (A.26)$$

Hence :

$$\frac{dy}{dS} = \frac{dy}{d\phi}\frac{d\phi}{dS} \qquad \frac{dx}{dS} = \frac{dx}{d\phi}\frac{d\phi}{dS}$$

$$dS(z) = \sqrt{\left(\frac{\partial x}{\partial \phi}\right)^2 + \left(\frac{\partial y}{\partial \phi}\right)^2}\; d\phi(z) \tag{A.27}$$

Similarity :

$$dS(\zeta) = \sqrt{\left(\frac{\partial \xi}{\partial \phi}\right)^2 + \left(\frac{\partial \eta}{\partial \phi}\right)^2}\; d\phi(\zeta) \tag{A.28}$$

A new variable will be introduced :

$$\gamma'(\zeta) = \gamma(\zeta)\sqrt{\left(\frac{\partial \xi}{\partial \phi}\right)^2 + \left(\frac{\partial \eta}{\partial \phi}\right)^2} \tag{A.29}$$

This definition of the vortex strength has no physical significance, it is just used to simplify the derivation. This definition also applies to the basic flow vortex strengths.

The singular integral

Expressions (A.15) through (A.18) are singular integrals. This computational difficulty can be avoided by finding a function whose derivative is equal to the integral. Thus the integrals will become equivalent to the function evaluated between the limits of integration. The function $I(z,\zeta)$ must be determined such that :

$$\frac{dI(z,\zeta)}{d\phi(\zeta)} = \left[f_y(z,\zeta)\,\frac{dx}{d\phi} - f_x(z,\zeta)\,\frac{dy}{d\phi}\right] \tag{A.30}$$

Make the following assumptions :

(1) $\gamma'(\zeta)$ is constant over the distance $\phi - \frac{\Delta\phi}{2}$ to $\phi + \frac{\Delta\phi}{2}$ so that the vortex strength in (A.15) through (A.18) can be taken out from under the integral sign.

(2) Also assume that $\frac{\partial x}{\partial \phi(\gamma)}$ and $\frac{\partial y}{\partial \phi(\zeta)}$ are constant in that region.

Define the following two functions :

$$F(z,\zeta) = \operatorname{atan} \frac{X}{Y} \qquad\qquad G(z,\zeta) = \frac{1}{2} \ln (X^2+Y^2)$$

Hence :

$$\frac{dG(z,\zeta)}{d\phi(\zeta)} = \frac{-X \dfrac{d\xi}{d\phi(\zeta)} - Y \dfrac{d\eta}{d\phi(\zeta)}}{X^2+Y^2}$$

$$\frac{dF(z,\zeta)}{d\phi(\zeta)} = \frac{X \dfrac{d\eta}{d\phi(\zeta)} - Y \dfrac{d\xi}{d\phi(\zeta)}}{X^2+Y^2}$$

Now multiply $\frac{dF}{d\phi(\zeta)}$ by C_1 and $\frac{dG}{d\phi(\zeta)}$ by C_2 where :

$$C_1 = \frac{\dfrac{d\xi}{d\phi(\zeta)} \dfrac{dx}{d\phi(z)} - \dfrac{dx}{d\phi(z)} \dfrac{d\eta}{d\phi(\zeta)}}{\left(\dfrac{d\xi}{d\phi(\zeta)}\right)^2 + \left(\dfrac{d\eta}{d\phi(\zeta)}\right)^2}$$

$$C_2 = \frac{\dfrac{d\xi}{d\phi(\zeta)} \dfrac{dx}{d\phi(z)} + \dfrac{dy}{d\phi(z)} \dfrac{d\eta}{d\phi(\zeta)}}{\left(\dfrac{d\xi}{d\phi(\zeta)}\right)^2 + \left(\dfrac{d\eta}{d\phi(\zeta)}\right)^2}$$

After simplification, it can be shown that the function $I(z,\zeta)$ in (A.30) has been obtained.

$$\frac{dI(z,\zeta)}{d\phi(\zeta)} = C_1 \frac{dF}{d\phi(\zeta)} + C_2 \frac{dG}{d\phi(\zeta)} \tag{A.31}$$

$$I(z,\zeta) = C_1 F(z,\zeta) + C_2 G(z,\zeta) \tag{A.32}$$

Hence the integral equations (A.15) through (A.18) reduce to the following summations :

$$\frac{1}{2\pi} \sum_{i=1}^{N} \alpha'(\zeta_i) K(z,\zeta_i,\Delta\zeta_i) = \frac{dy_j}{d\phi} \qquad (A.33)$$

$$\frac{1}{2\pi} \sum_{i=1}^{N} \beta'(\zeta_i) K(z,\zeta_i,\Delta\zeta_i) = \frac{dx_j}{d\phi} \qquad (A.34)$$

$$\frac{1}{2\pi} \sum_{i=1}^{N} \mu'(\zeta_i) K(z,\zeta_i,\Delta\zeta_i) = 0 \qquad (A.35)$$

$$\frac{1}{2\pi} \sum_{i=1}^{N} \delta'(\zeta_i) K(z,\zeta_i,\Delta\zeta_i) \qquad (A.36)$$

$$= - \frac{1}{2\pi} \oint Q'(\zeta) \left[f_x(z,\zeta) \frac{dx}{d\phi} - f_y \frac{dy}{d\phi} \right] dS(\zeta)$$

where :

$$K(z,\zeta_i,\Delta\zeta_i) = I \left(z,\zeta_i + \frac{\Delta\zeta_i}{2} \right) - I \left(z,\zeta_i - \frac{\Delta\zeta_i}{2} \right) \qquad (A.37)$$

The supplementary conditions are :

$$\sum_{i=1}^{N} \alpha(\zeta_i) \Delta S(\zeta_i) = 0 \qquad (A.38)$$

$$\sum_{i=1}^{N} \beta(\zeta_i) \Delta S(\zeta_i) = 0 \qquad (A.39)$$

$$\sum_{i=1}^{N} \mu(\zeta_i) \Delta S(\zeta_i) = 1 \qquad (A.40)$$

$$\sum_{i=1}^{N} \delta(\zeta_i) \Delta S(\zeta_i) = 0 \qquad (A.41)$$

CFC1. FOR

```
C****************************************************************
C****************************************************************
C                 CFC1  MAIN PROGRAM
C****************************************************************
C   PURPOSE
C   THIS PROGRAM CALCULATES THE CONTROL POINT AND SINGULARITY POINT
C   LOCATIONS FOR AN AIRFOIL.
C****************************************************************
C   DISCRIPTION OF PARAMETERS
C          N    -NUMBER OF POINTS WHERE THE VELOCITY IS TO BE CALCULATED
C         NP    -NUMBER OF AIRFOIL POINTS READ IN
C         EL    -STAGGER ANGLE                             (DEG.)
C          T    -PITCH TO CHORD RATIO
C      PROFL    -PROFILE DESIGNATION
C        X,Y    -TRANSFORMED COORDINATE POINT LOCATIONS ON THE PROFILE
C      XP,YP    -DX/DPHI, DY/DPHI RESPECTIVELY
C        PHI    -THE TRANSFORMED COORDINATE
C   XI,YT,YC    -THE PROFILE THICKNESS AND CAMBERLINE DISTRIBUTION
C         RC    -THE RADIUS OF CURVATURE
C         B1     -THE INLET AIR ANGLE (ANGLE OF ATTACK)    (DEG.)
C****************************************************************
C   REMARKS
C   THIS PROGRAM IS TO BE USED WITH TWO OTHER PROGRAMS (CFC2, AND SFC3)
C   ALTHOUGH IT IS A COMPLETE UNIT AS IT STANDS.
C   THE FOLLOWING STEPS ARE PERFORMED IN THE CALCULATION.
C       1.) THE INPUT DATA IS THE AIRFOIL GEOMETRY AND IS READ IN
C           AS AN ARRAY OF THE PROFILE COORDINATE LOCATIONS OR AS A
C           THICKNESS AND CAMBERLINE DISTRIBUTION
C       2.) THE CONTROL AND SINGULARITY POINT LOCATIONS ARE
C           DETERMINED BY TRANSFORMING THE COORDINATE PROFILE AND
C           THEN DIVIDING THE TRANSFORMED CONTOUR INTO EQUAL
C           SEGMENTS.  THE VARIABLE NP DETERMINES THE TOTAL NUMBER
C           OF POINTS (NP CONTROL AND NP SINGULARITY POINTS).
C       3.) THE DERIVATIVES DX/DPHI AND DY/DPHI ARE ALSO CALCULATED
C           WHERE PHI IS THE TRANSFORMED COORDINATE.
C   THE TRANSFORMED COORDINATES AND DERIVATIVES ARE THEN WRITTEN ON THE
C   DISK FOR USE BY PROGRAMS CFC2, AND SFC3.
C   THE PROGRAM OPTIONS ARE GOVERNED BY DATA SWITCHES WHICH ARE READ. THE
C   OPTIONS ARE
C               1.)  DATA SWITCH 1  ON, THE PROFILE COORDINATES ARE READ
C               2.)  DATA SWITCH 2  ON, THE CALCULATION STARTS WITH
C                                       THE DERIVATIVES ALREADY DETERMINED
C               3.)  DATA SWITCH 3  ON, THE CAMBERLINE AND THICKNESS
C                                       DISTRIBUTION IS READ
C               4.)  DATA SWITCH 4  ON, SMOOTHING IS NOT PERFORMED
C                                  OFF, SMOOTHING IS DONE
C               5.)  DATA SWITCH 16 ON, THE AIRFOIL CALCULATION IS DONE
C                                  OFF, THE CASCADE CALCULATION IS DONE
C   IF DATA SWITCHES 1-3 ARE OFF, THEN A SET OF ARBITRARILY LOCATED
C   POINTS SHOULD BE READ.
C****************************************************************
      COMMON /YGSW/ I(19)
      INTEGER IP(40)
      DO 4 J=1,19
4     I(J)=2
C   READ THE PROGRAM OPTIONS (DATA SWITCHES)
C   IF A DATA SWITCH IS READ, IT IS TURNED ON.
      READ(2,1) IP
      DO 2 J=1,40
      K=IP(J)
      IF(K .LT. 0 .OR. K .GT. 18) GOTO 3
      IF (K .EQ. 0) K=19
2     I(K)=1
3     CONTINUE
      WRITE(3)(I(K), K=1,19)
      CALL CFCM1
1     FORMAT(40I2)
      STOP
      END
C****************************************************************
      SUBROUTINE CFCM1
      COMMON/ACFC/NP,N2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      DIMENSION XPLI(480), YPLI(480)
      PI=3.14159265 3
      CALL DATSW(1,JA1)
      IF (JA1 .EQ. 2) GOTO 7
      CALL CFCCZ
      GO TO 5
    7 CONTINUE
      CALL DATSW(2,JA)
      IF(JA .EQ. 1) GO TO 1
      CALL DATSW(3,JB)
      IF(JB .EQ. 2) GOTO 4
      CALL CFCCA
      GO TO 5
    4 CONTINUE
      CALL CFCCF
    5 CONTINUE
      CALL CFC A
      CALL CFCCC
    1 CONTINUE
      N2=NP*2
C   THE TRANSFORMED CONTOUR COORDINATE AND FIRST DERIVATIVES ARE
C   STORED  ON USE BY PROGRAMS CFC2 AND SFC3.
      WRITE(3) NP, N2, T, EL, B1R
      WRITE(3) (PROFL(I), I=1,10)
      WRITE(3) (X(I), I=1,N2)
      WRITE(3) (Y(I), I=1,N2)
      WRITE(3) (XP(I), I=1,N2)
      WRITE(3) (YP(I), I=1,N2)
      WRITE(3) (RC(I), I=1,N2)
      RETURN
      END
C****************************************************************
      SUBROUTINE INTPO(NI,XI,X,F)
      DIMENSION X(1),F(1)
      J=0
      NP=NI
    1 NP=NP-1
      J=J+1
      DO 2 K=1,NP
      I=K+J
    2 F(K)=(F(K)*(X(I)-XI)+F(K+1)*(XI-X(K)))/(X(I)-X(K))
      IF(NP-1)3,3,1
    3 RETURN
      END
C****************************************************************
      SUBROUTINE INTCF(P,W1,W2,A3,K3,NZ)
      COMMON/CCFC/YC(480),YT(480),PHI(480),XI(480),RC(480)
      DIMENSION W(3,6),S(6)
C   THIS SUBROUTINE PERFORMS LAGRANGE INTERPOLATION.
```

```
      K5=-1
      NZ1=NZ-1
      IF(PHI(2)-P)10,20,20
   10 IF(P-PHI(NZ1))30,28,28
   28 K3=NZ-3
      K5=1
   20 N3=3
      GOTO40
   30 N3=4
   40 N1=N3
      N5=N3-1
   50 DO21J=1,N3
      L1=J+K3
      W(1,J)=YC(L1)
      W(2,J)=YT(L1)
      W(3,J)=XT(L1)
   21 S(J)=PHI(L1)
      J1=0
      IF(K5)22,23,23
   22 IF(S(N5)-P)24,23,23
   24 K3=K3+1
      GOTO50
   23 N1=N1-1
      J1=J1+1
      DO25K=1,N1
      L=K+J1
      DO26M=1,3
   26 W(M,K)=(W(M,K)*(S(L)-P)+W(M,K+1)*(P-S(K)))/(S(L)-S(K))
   25 CONTINUE
      IF(N1-1)27,27,23
   27 W1=W(1,1)
      W2=W(2,1)
      W3=W(3,1)
      RETURN
      END
C***********************************************************************
      SUBROUTINE DERCF(N,DER)
      COMMON/CCFC/W(480),DUM(960),S(480),RC(480)
      DIMENSION Y(5),X(5),XX(5),YY(5),A1(5),A2(3),A3(2),DER(260)
      M=0
      K=1
      M1=0
      I1=0
      JJ=1
   17 DO 10 I=JJ,5
      L=I+K-1
      XX(I)=S(L)
   10 YY(I)=W(L)
   16 M=M+1
      M1=M-L+5
      X(1)=XX(M1)
      Y(1)=YY(M1)
      M3=M1-1
      DO 1 I=1,M3
      IF(I-M1)18,1,18
   18 K1=M1-I
      X(K1+1)=XX(K1)
      Y(K1+1)=YY(K1)
    1 CONTINUE
      M2=M1+1
      IF(M2-6)20,21,21
   20 DO 19 I=M2,5
      X(I)=XX(I)
   19 Y(I)=YY(I)
   21 DO 2 I=1,4
    2 A1(I)=(Y(I)-Y(I+1))/(X(I)-X(I+1))
      DO 3 I=1,3
    3 A2(I)=(A1(I)-A1(I+1))/(X(I)-X(I+2))
      DO 4 I=1,2
    4 A3(I)=(A2(I)-A2(I+1))/(X(I)-X(I+3))
      A4=(A3(1)-A3(2))/(X(1)-X(5))
      I1=I1+1
      DER(I1)=A1(1)-(X(2)-X(1))*A2(1)+(X(2)-X(1))*(X(3)-X(1))*A3(1)-(X(2
     1)-X(1))*(X(3)-X(1))*(X(4)-X(1))*A4
      JJ=3
      IF(M-2)22,22,23
   23 IF(M-N+2)23,22,22
   22 IF(K-N)26,27,27
   26 GO TO 16
   24 DO 25 I=1,2
      XX(I)=XX(I+1)
   25 YY(I)=YY(I+1)
      K=K+1
      GO TO 17
   27 RETURN
      END
C***********************************************************************
      SUBROUTINE CFCCC
      COMMON/ACFC/NP,N2,T,EL,PROFL(10),BIP,PT
      COMMON/BCFC/X(516),Y(516),X2(508),Y2(508),XT1,X1(480),Y1(480),
     1DUM(484)
      COMMON/CCFC/XF(480),YF(480),XP(480),YP(480),RC(480)
C  COMPUTE THE RADIUS OF CURVATURE.
      N22=2*NP
      DO 20 I=1,N22
      RC(I)=(X1(I)**2+Y1(I)**2)**1.5/(X1(I)*Y2(I)-X2(I)*Y1(I))
   20 CONTINUE
C  OUTPUT OF BLADE COORDINATES
      WRITE(1,102)
      WRITE(1,1) PROFL
      ND1=NP/2+1
      NV1=NP-1-1
      WRITE(1,91)
      DO1103 K=1,ND1
      J=2*K+NV1
      XS=X(J)*XT1
      YS=Y(J)*XT1
 1103 WRITE(1,100)XS,YS ,X1(J),Y1(J),X2(J),Y2(J),RC(J)
      ND1=NP/2-1
      NV1=0
      DO1105 K=1,ND1
      J=2*K+NV1
      XS=X(J)*XT1
      YS=Y(J)*XT1
```

```
1105 WRITE(1,100)XS,YS ,X1(J),Y1(J),X2(J),Y2(J),RC(J)
     ND1=NE/2
     NV1=NP-1
     DO1106 K=1,ND1
     J=2*K+NV1
     XS=X(J)*XT1
     YS=Y(J)*XT1
1106 WRITE(1,100)XS,YS ,X1(J),Y1(J),X2(J),Y2(J),RC(J)
     NV1=0-1
     DO1107 K=1,ND1
     J=2*K+NV1
     XS=X(J)*XT1
     YS=Y(J)*XT1
1107 WRITE(1,100)XS,YS ,X1(J),Y1(J),X2(J),Y2(J),RC(J)
     ND1=NP/2+1
     NV1=NP-1-1
     KF=0
     DO 103 K=1,ND1
     J=2*K+NV1
     KF=KF+1
     XF(KF)=X(J)
     YF(KF)=Y(J)
     XP(KF)=X1(J)
 103 YP(KF)=Y1(J)
     ND1=NP/2-1
     NV1=0
     DO 105 K=1,ND1
     J=2*K+NV1
     KF=KF+1
     XF(KF)=X(J)
     YF(KF)=Y(J)
     XP(KF)=X1(J)
 105 YP(KF)=Y1(J)
     ND1=NP/2
     NV1=NP-1
     DO 106 K=1,ND1
     J=2*K+NV1
     KF=KF+1
     XF(KF)=X(J)
     YF(KF)=Y(J)
     XP(KF)=X1(J)
 106 YP(KF)=Y1(J)
     NV1=0-1
     DO 107 K=1,ND1
     J=2*K+NV1
     KF=KF+1
     XF(KF)=X(J)
     YF(KF)=Y(J)
     XP(KF)=X1(J)
 107 YP(KF)=Y1(J)
   1 FORMAT(1X,10A4)
  91 FORMAT(33H BLADE COORDINATES AND DERIVATIVES,/,7X,1HX,12X,1HY,12X,
    12HXP,11X,2HYP,11X,2HXS,11X,2HYS,11X,'RC',/)
 100 FORMAT(1X,7F13.6)
 102 FORMAT(1H1)
     RETURN
     END
C*************************************************************************
     SUBROUTINE CFCCF
     COMMON/ACFC/NF,N2,T,EL,PROFL(10),B1R,PI
     COMMON/BCFC/X(516),Y(516),DUM1(1016),XC,DUM2(1444)
```

```
C   CALCULATION OF PROFILE COORDINATES, FIRST AND SECOND DERIVATIVE
C   STARTING FROM AN ARBRITARY LOCATED SET OF BLADE COORDINATES
      READ(2,21)PROFL
      READ(2,1000)I,EL
      CALL DATSW(16,JAIR)
      IF(JAIR.EQ.1) T=2.0*PI
      READ(2,1001)NP
      N=2*NP+4
      DO 10 K=5,N
   10 READ(2,102)X(K),Y(K)
      NPL=5+NP
      XL=X(NPL)
      YL=Y(NPL)
      DO 50 I=5,N
      X(I)=X(I)-XL
 50   Y(I)=Y(I)-YL
      ALF=ATAN(Y(5)/X(5))
      SALF=SIN(ALF)
      CALF=COS(ALF)
      DO 51 I=5,N
      XT=Y(I)*SALF+X(I)*CALF
      YT=Y(I)*CALF-X(I)*SALF
      X(I)=XT
 51   Y(I)=YT
      EL=EL+57.2958279*ALF
      XC=X(5)
      DO 30 K=5,N
      X(K)=X(K)/XC
   30 Y(K)=Y(K)/XC
      X(N+1)=1
      Y(N+1)=0.
      DO 20 K=1,4
      NN=N-4+K
      X(K)=X(NN)
      Y(K)=Y(NN)
      NN=N+1+K
      X(NN)=X(K+5)
   20 Y(NN)=Y(K+5)
   21 FORMAT(10A4)
  102 FORMAT(2F10.7)
 1000 FORMAT(F9.6,F7.3)
 1001 FORMAT(I2)
      RETURN
      END
C*************************************************************************
      SUBROUTINE CFCCZ
      COMMON/ACFC/NF,N2,T,FL,PROFL(10),B1R,PI
      COMMON/BCFC/XE(516),YE(516),XI(400),YI(400),DUMQ(216),C,SF(400),
     1SI(400),SEE(244),SII(244),DUMB(156)
      COMMON/CCFC/XEE(244),YEE(244),XII(244),YII(244),DUMY(944),RC(480)
      DIMENSION A(6),FX(6),FY(6)
C   CALCULATION OF BLADE FIRST AND SECOND DERIVATIVES STARTING FROM
C   THE PROFILE COORDINATES.
      READ(2  ,100) PROFL
      READ(2  ,200) T,FL
      CALL DATSW(16,JAIR)
      IF(JAIR.EQ.1) T=2.0*PI
      READ( 2 ,1) NI,NP
      READ( 2 ,2) (XE(I),YE(I),XI(I),YI(I),I=1,NI)
      IF(XE(1))4,3,4
    3 IF(YE(1))4,6,4
```

```
    4 XE1=XE(1)
C   TRANSLATE THE PROFILE COORDINATES.
      YE1=YE(1)
      XI1=XI(1)
      YI1=YI(1)
      DO 5 I=1,NI
      XE(I)=XE(I)-XE1
      YE(I)=YE(I)-YE1
      XI(I)=XI(I)-XI1
    5 YI(I)=YI(I)-YI1
C   ROTATE THE PROFILE COORDINATES.
    6 AL=ATAN(YE(NI)/XE(NI))
      EL=EL+57.2958279*AL
      SAL=SIN(AL)
      CAL=COS(AL)
      DO 7 I=1,NI
      XEC=XE(I)
      YEC=YE(I)
      XIC=XI(I)
      YIC=YI(I)
      XE(I)=XEC*CAL+YEC*SAL
      YE(I)=YEC*CAL-XEC*SAL
      XI(I)=XIC*CAL+YIC*SAL
    7 YI(I)=YIC*CAL-XIC*SAL
      C=XE(NI)
      DO 8 I=1,NI
      XE(I)=XE(I)/C
      YE(I)=YE(I)/C
      XI(I)=XI(I)/C
    8 YI(I)=YI(I)/C
      SE(1)=0.
      SI(1)=0.
      DO 9 I=2,NI
      SE(I)=SQRT((XE(I)-XE(I-1))**2+(YE(I)-YE(I-1))**2)+SE(I-1)
      SI(I)=SQRT((XI(I)-XI(I-1))**2+(YI(I)-YI(I-1))**2)+SI(I-1)
    9 CONTINUE
      SSE=SE(NI)
      SSI=SI(NI)
C   CALCULATE THE TRANSFORMED PROFILE COORDINATES.
      NP1=NP+1
      CT=PI/FLOAT(NP)
      DO 10 J=1,NP1
      FI=FLOAT(J-1)*CT
      CF=COS(FI)
      SF=SIN(FI)
      SEE(J)=SSE*(1.-CF)/2.
      R=SQRT(1./(CF*CF+SF*SF*SSE/SSI))
   10 SII(J)=SSI*(1.-R*CF)/2.
      DO 41 NC=1,2
      NI1=NI-1
      NI2=NI-2
      NI3=NI+1
      DO 40 J=2,NP
      DO 13 I=1,NI
      IF(NC .EQ. 2) GOTO 12
      IF(SE(I).GT.SEE(J)) GO TO 14
      GO TO 13
   12 IF(SI(I).GT.SII(J)) GO TO 14
   13 CONTINUE
      GO TO 40
   14 J6=I+2
      IF(I.GE.4) GO TO 19
      I6=IABS(J6-6)
      N=I6+1
      DO 17 K=1,I6
      IF(NC .EQ. 2) GOTO 16
      A(K)=-SE(N)
      FX(K)=-XE(N)
      FY(K)=-YE(N)
      GO TO 17
   16 A(K)=-SI(N)
      FX(K)=-XI(N)
      FY(K)=-YI(N)
   17 N=N-1
      I6=I6+1
      N=1
      GO TO 20
   19 IF(I.GT.NI2) GO TO 25
      N=J6-5
      I6=1
   20 DO 23 K=I6,6
      IF(NC .EQ. 2) GOTO 22
      A(K)=SE(N)
      FX(K)=XE(N)
      FY(K)=YE(N)
      GO TO 23
   22 A(K)=SI(N)
      FX(K)=XI(N)
      FY(K)=YI(N)
   23 N=N+1
      GO TO 34
   25 N=I-3
      I6=NI3-N
      DO 28 K=1,I6
      IF(NC .EQ. 2) GOTO 27
      A(K)=SE(N)
      FX(K)=XE(N)
      FY(K)=YE(N)
      GO TO 28
   27 A(K)=SI(N)
      FX(K)=XI(N)
      FY(K)=YI(N)
   28 N=N+1
      I6=I6+1
      N=NI1
      DO 32 K=I6,6
      IF(NC .EQ. 2) GOTO 31
      A(K)=2.*SE(NI)-SE(N)
      FX(K)=-XE(N)+2.*XE(NI)
      FY(K)=-YE(N)
      GO TO 32
   31 A(K)=2.*SI(NI)-SI(N)
      FX(K)=-XI(N)+2.*XI(NI)
      FY(K)=-YI(N)
   32 N=N-1
34    IF(NC .EQ. 2) GOTO 39
      AI=SEE(J)
      CALL INTIP(6,AI,A,FX)
      XEE(J)=FX(1)
      CALL INTIP(6,AI,A,FY)
      YEE(J)=FY(1)
      GO TO 40
```

```
   39 AI=SII(J)
      CALL INTPO(6,AI,A,FX)
      XII(J)=FX(1)
      CALL INTPO(6,AI,A,FY)
      YII(J)=FY(1)
   40 CONTINUE
   41 CONTINUE
      NP3=NP+1
      XEE(1)=XE(1)
      YEE(1)=YE(1)
      XEE(NP3)=XE(NI)
      YEE(NP3)=YE(NI)
      XII(1)=XI(1)
      YII(1)=YI(1)
      XII(NP3)=XI(NI)
      YII(NP3)=YI(NI)
      NI=NP3/2
      NR=MOD(NP3,2)
      NT2=NI*2
      NTR=NI+NR
      XE(NTR)=XEE(NTR)
      YE(NTR)=YEE(NTR)
      DO 42 NC=1,NI
      NF=NT2-NC+NR+1
      XE(NC)=XEE(NF)
      YE(NC)=YEE(NF)
      XE(NF)=XEE(NC)
   42 YE(NF)=YEE(NC)
      NPI=NP+2
      NPF=NP*2
      DO 43 I=NPI,NPF
      K=I-NP
      XE(I)=XII(K)
   43 YE(I)=YII(K)
      KPI=NPF+5
      KPF=NPF+9
      DO 44 I=KPI,KPF
      K=I-KPI+1
      XE(I)=XE(K)
   44 YE(I)=YE(K)
      DO 45 I=1,NPF
      K=NPF+5-I
      L=NPF+1-I
      XE(K)=XE(L)
   45 YE(K)=YE(L)
      DO 46 I=1,4
      K=NPF+I
      XE(I)=XE(K)
   46 YE(I)=YE(K)
      NPF=NPF+8
    1 FORMAT(2I3)
    2 FORMAT(3F10.5)
  100 FORMAT(10A4)
  199 FORMAT(5X,I5,2F12.5)
  200 FORMAT(F9.6,F7.3)
      RETURN
      END
C*********************************************************************
      SUBROUTINE CFCCA
      COMMON/ACFC/NC,N2,T,EL,PROFL(10),DIR,PI
      COMMON/BCFC/X(516),Y(516),X2(508),Y2(508),XT1,X1(480),Y1(480),
     1DUM(481)
      COMMON/CCFC/YC(480),YT(480),PHI(480),XT(480),RC(480)
      DIMENSION DER(260)
C     CALCULATION OF PROFILE COORDINATES, FIRST AND SECOND DERIVATIVE
C     STARTING FROM CAMBERLINE AND THICKNESS DISTRIBUTION
      READ(2,11)PROFL
      READ(2,1000)T,EL
      CALL DATSW(16,JAIR)
      IF(JAIR.EQ.1) T=2.0*PI
      READ(2,1001)NP,NZ
      READ(2,1002)(YC(J),YT(J),XT(J),J=1,NZ)
      XT1=XT(1)
      DO 1004J=1,NZ
      YC(J)=YC(J)/XT1
      YT(J)=YT(J)/XT1
      XT(J)=XT(J)/XT1
 1004 CONTINUE
      YT(1)=0.
      READ(2,1003)SFC,SFT
      DO 121 I=1,NZ
      YC(I)=YC(I)*SFC
      YT(I)=YT(I)*SFT
  121 CONTINUE
      I=1
      K3=0
      DER=0.
      PHI(1)=0.
      PHI(NZ)=PI
      NZ1=NZ-1
      DO 6 J=2,NZ1
      IF(XT(J)-.5)2,4,5
    4 PHI(J)=PI*.5
      GO TO 6
    2 XT2=1.-2.*XT(J)
      PHI(J)=PI-ATAN((SQRT(1.-XT2*XT2))/XT2)
      GO TO 6
    5 XT2=2.*XT(J)-1.
      PHI(J)=ATAN((SQRT(1.-XT2*XT2))/XT2)
    6 CONTINUE
      NN=NP-1
      DO7N=1,NN
      AN=N
      ANP=NP
      E2=AN*PI/ANP
      CALL INICF(E2,W1,W2,W3,K3,NZ)
      Y1(N+240)=W1
      Y1(N)=W2
    7 X1(N)=W3
      DO3N=1,NN
      NNM3=NN-N+4
      YC(NNM3)=Y1(N+240)
      YT(NNM3)=Y1(N)
    3 XT(NNM3)=X1(N)
      DO8MM=1,2
      NNM4=6-MM
      YC(MM)=-YC(NNM4)
      YT(MM)=-YT(NNM4)
      XT(MM)=-XT(NNM4)
      NNM1=NN+MM+4
      NNM2=NN-MM+4
      YT(NNM1)=-YT(NNM2)
```

```
      YC(NNM1)=-YC(NNM2)
    8 XT(NNM1)=2.-XT(NNM2)
      YC(3)=0.
      YT(3)=0.
      XT(3)=0.
      YT(NN+1)=0.
      YC(NN+4)=0.
      XT(NN+1)=1.
      NN=NP+4
      CALL DEPCF(NN,DER)
      NN=NP+1
      DO9N=5,NN
      NNN=NP-N+8
      ATA=ATAN(DER(NNN))
      Y(N)=YC(NNN)+YT(NNN)*COS(ATA)
      X(N)=XT(NNN)-YT(NNN)*SIN(ATA)
      NM=2*NP-N+10
      Y(NM)=YC(NNN)-YT(NNN)*COS(ATA)
    9 X(NM)=XT(NNN)+YT(NNN)*SIN(ATA)
      Y(NN+1)=0.
      X(NN+1)=0.
      Y(5)=0.
      X(5)=1.
      NN=2*NP+5
      Y(NN)=0.
      X(NN)=1.
      DO10N=1,4
      NNM5=2*NP+N
      Y(N)=Y(NNM5)
      X(N)=X(NNM5)
      NNM1=NN+N
      NNM2=5+N
      Y(NNM1)=Y(NNM2)
   10 X(NNM1)=X(NNM2)
   11 FORMAT(10A4)
 1000 FORMAT(F9.6,F7.3)
 1001 FORMAT(2I3)
 1002 FORMAT(3F9.6)
 1003 FORMAT(2F5.3)
      RETURN
      END
C*********************************************************************
      SUBROUTINE CFCCB
      COMMON/ACFC/NP,N2,T,EL,PRCFL(10),BIR,PI
      COMMON/BCFC/X(516),Y(516),X2(508),Y2(508),XT1,X1(480),Y1(480),
     1DUM(194)
      NN=2*NP-1
      CALL DATSW(3,JA)
      IF(JA .EQ. 1) GOTO 10
      DO 15 N=3,NN
      Y2(N)=(-Y(N-2)-Y(N+2)+4.*(Y(N-1)+Y(N+1))+6.*Y(N))/12.
   15 X2(N)=(-X(N-2)-X(N+2)+4.*(X(N-1)+X(N+1))+6.*X(N))/12.
      DO 21 N=3,NN
      Y(N)=Y2(N)
   21 X(N)=X2(N)
   10 CONTINUE
      PIN1=NP
      PIN1=PIN1/(12.*PI)
      PRN=NP*NP
      PRN=PRN/(7.*PI*PI)
      PIN2=PIN1**2
      DO16N=1,NN
      NM=N+5
      Y1(N)=(Y(NM-2)-8.*Y(NM-1)+8.*Y(NM+1)-Y(NM+2))*PIN1
      X1(N)=(X(NM-2)-8.*X(NM-1)+8.*X(NM+1)-X(NM+2))*PIN1
      Y2(N)=(Y(NM-4)-16.*Y(NM-3)+64.*Y(NM-2)+16.*Y(NM-1)-130.*
     1Y(NM)+16.*Y(NM+1)+64.*Y(NM+2)-16.*Y(NM+3)+Y(NM+4))*PIN2
      X2(N)=(X(NM-4)-16.*X(NM-3)+64.*X(NM-2)+16.*X(NM-1)-130.*
     1X(NM)+16.*X(NM+1)+64.*X(NM+2)-16.*X(NM+3)+X(NM+4))*PIN2
   16 CONTINUE
      Y1(NN+1)=(Y(3)-8.*Y(4)+8.*Y(6)-Y(7))*PIN1
      X1(NN+1)=(X(3)-8.*X(4)+8.*X(6)-X(7))*PIN1
      Y2(NN+1)=(Y(1)-16.*Y(2)+64.*Y(3)+16.*Y(4)-130.*Y(5)+16
     1.*Y(6)+64.*Y(7)-16.*Y(8)+Y(9))*PIN2
      X2(NN+1)=(X(1)-16.*X(2)+64.*X(3)+16.*X(4)-130.*X(5)+16
     1.*X(6)+64.*X(7)-16.*X(8)+X(9))*PIN2
      NN=NN+1
      DO17N=1,NN
      Y(N)=Y(N+5)
   17 X(N)=X(N+5)
      RETURN
      END
C*********************************************************************
      SUBROUTINE DATSW(I,J)
      COMMON/YGSW/ K(19)
      IL=I
      IF(IL.EQ.0.OR.IL.GT.18) IL=19
      J=K(IL)
      RETURN
      END
C*********************************************************************
```

CFC2.FOR

```
C************************************************************
C************************************************************
C                   CFC2   MAIN PROGRAM
C************************************************************
C   PURPOSE
C   THIS PROGRAM CALCULATES THE INVISID, POTENTIAL FLOW OVER A 2-D
C   AIRFOIL.
C************************************************************
C   DISCRIPTION OF PARAMETERS
C            N   -NUMBER OF POINTS WHERE THE VELOCITY IS TO BE CALCULATED
C           NF   -NUMBER OF AIRFOIL POINTS READ IN
C           EL   -STAGGER ANGLE                               (DEG.)
C            T   -PITCH TO CHORD RATIO
C        PROFL   -PROFILE DESIGNATION
C          X,Y   -TRANSFORMED COORDINATE POINT LOCATIONS ON THE PROFILE
C        XP,YP   -DX/DPHI, DY/DPHI RESPECTIVELY
C          PHI   -THE TRANSFORMED COORDINATE
C     XT,YT,YC   -THE PROFILE THICKNESS AND CAMBERLINE DISTRIBUTION
C           RC   -THE RADIUS OF CURVATURE
C           B1   -THE INLET AIR ANGLE (ANGLE OF ATTACK)      (DEG.)
C           CU   -THE VORTEX STRENGTH AND VELOCITY
C           CO   -THE COEFFICIENTS OF THE K MATRIX
C          COL   -COLUMNS IN THE K MATRIX
C************************************************************
C   REMARKS
C   THIS PROGRAM IS TO BE USED WITH TWO OTHER PROGRAMS (CFC1, AND SFC3)
C   ALTHOUGH IT IS A COMPLETE UNIT AS IT STANDS, IT REQUIRES THE RESULTS
C   OF CFC1.
C   THE PROGRAM OPTIONS ARE GOVERNED BY DATA SWITCHES WHICH ARE READ IN
C   PROGRAM CFC1. THE OPTIONS ARE
C               1.)  DATA SWITCH 10 ON, A DIFFERENT POINT OF APPLICATION
C                                   OF THE KUTTA CONDITION IS POSSIBLE
C               2.)  DATA SWITCH 12 ON,  THE CIRCULATION MAY BE READ
C                                  OFF,  THE CIRCULATION IS CALCULATED
C               3.)  DATA SWITCH 15 ON, INTERMEDIATE RESULTS ARE PRINTED.
C                                  OFF, NO INTERMEDIATE RESULTS PRINTED.
C               4.)  DATA SWITCH 16 ON, THE AIRFOIL CALCULATION IS DONE
C                                  OFF, THE CASCADE CALCULATION IS DONE
C   THE RESULTS OF THIS PROGRAM ARE WRITTEN ON THE DISK FOR USE BY
C   PROGRAM SFC3.
C************************************************************
      COMMON /YSSS/ ISWITCH(19)
      COMMON/CFC92/ Y92(240,240)
      COMMON/CFC83/ Y83(480,7)
      COMMON/CFC91/ Y91(480,240)
      COMMON/CFC99/ Y99(10,2)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/FCFC/ XSAVE(240), CUSAVE(240)
      READ(3) (ISWICH(K), K=1,19)
      READ(3) N, N2, T, EL, B1R
      READ(3) (PROFL(I), I=1,10)
      READ(3) (X(I), I=1,N2)
      READ(3) (Y(I), I=1,N2)
      READ(3) (XP(I), I=1,N2)
      READ(3) (YP(I), I=1,N2)
      READ(3) (RC(I), I=1,N2)
      CALL CFCM2
      WRITE(3) ((Y92(I,J), I=1,240), J=1,240)
      WRITE(4) ((Y91(I,J), I=1,480), J=1,240)
      WRITE(10) (XSAVE(I), I=1,N)
      WRITE(10) (CUSAVE(I), I=1,N)
      STOP
      END
C************************************************************
      SUBROUTINE CFCM2
      CALL CFC01
      CALL CFC02
      CALL CFC03
      CALL CFC04
      CALL CFC05
      RETURN
      END
C************************************************************
      SUBROUTINE CFC01
      COMMON/CFC83/ Y83(480,7)
      COMMON/CFC99/ Y99(10,2)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
C   READ PROFILE COORDINATES AT ZERO STAGGER.
C   READ GENERAL INPUT DATA.
      PI=3.14159
    9 CONTINUE
      WRITE(1,23)PROFL
      WRITE(1,2)T,EL,N
   12 N2=N*2
      T=2.0*PI/T
      DO 13 I=1,N
      Y83(I,6)=Y(I)
   13 Y83(I,7)=X(I)
      READ(2,80)B1
      B1R=PI*B1/180.0
      C1X=COS(B1R)
      C1Y=SIN(B1R)
      C2X=C1X
      C2Y=C1Y
      Y99(1,2)=C1X
      Y99(2,2)=C1Y
      Y99(3,2)=C2X
      Y99(4,2)=C2Y
C   PUT PROFILE COORDINATES AT THE GIVEN STAGGER ANGLE EL.
      ER=EL*PI/180.0
      CEL=COS(ER)
      SEL=SIN(ER)
      DO 15 I=1,N2
      YY1=Y(I)
      XX1=X(I)
      Y(I)=YY1*CEL+XX1*SEL
      X(I)=XX1*CEL-YY1*SEL
      YY1=YP(I)
      XX1=XP(I)
      YP(I)=YY1*CEL+XX1*SEL
   15 XP(I)=XX1*CEL-YY1*SEL
      WRITE(1,22)
      DO 7 I=1,N2
    7 WRITE(1,3)X(I),Y(I),XP(I),YP(I),RC(I)
    6 CONTINUE
    4 FORMAT(6F13.10)
   80 FORMAT(F7.3)
    1 FORMAT(2F10.5,I3)
    2 FORMAT(1X,2HT=,F9.5,3X,3HEL=,F9.5,3X,2HN=,I3)
 8    FORMAT(3X,5F9.4)
```

```
   22 FORMAT(1X,40HBLADE COORDINATES AT GIVEN STAGGER ANGLE/8X,1HX,8X,
     11HY,9X,2HXP,7X,2HYP,7X,'RC'/)
  23    FORMAT(1X,10A4)
        RETURN
        END
C*********************************************************************
        SUBROUTINE CFCO2
        COMMON/CFC91/ Y91(480,240)
        COMMON/ACFC/N,N2,T,EL,FPCFL(10),BIR,PI
        COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
        DIMENSION COL(480)
C   CALCULATE MATRIX COEFFICIENTS.
        N1=N-1
        T4P=T/(4.*PI)
        MT=N/2+1
        MTN=MT+N
        MTN1=MT+N-1
        AN=N
        DO 20 L=1,N
        IF(L-MT)10,11,12
  10    J=L
        GO TO 13
  11    GO TO 20
  12    J=L-1
  13    CONTINUE
        JCO=J
        DO 21 K=1,N2
        XC=XP(K)
        YC=YP(K)
        J2=L+N
        J1=L+N-1
        IF(J1-N)75,75,76
  75    J1=N2
  76    CONTINUE
        ADE=(XP(L)*XP(L)+YP(L)*YP(L))*T
        XC1= (XP(L)*YC-XC*YP(L))/ADE
        XC2= (XP(L)*XC+YP(L)*YC)/ADE
        ITEST=3
        IF(K-J1)18,17,18
   18   IF(K-J2)80,19,80
   17   JC=J1
        J1=L
        ITEST=1
        GO TO 80
   19   JC=J2
        J2=L
        ITEST=2
  80    CONTINUE
        XJ1=T*(X(K)-X(J1))/2.
        XJ2=T*(X(K)-X(J2))/2.
        YJ1=T*(Y(K)-Y(J1))/2.
        YJ2=T*(Y(K)-Y(J2))/2.
        XJ3=2.*XJ1
        XJ4=2.*XJ2
 100    YJ3=2.*YJ1
        YJ4=2.*YJ2
        YJ5=ABS(YJ1)-PI/2.0
        YJ6=ABS(YJ2)-PI/2.0
        IF(YJ5)101,102,102
 102    IF(YJ6)101,101,103
 103    IF(YJ1)104,104,105
 104    YJ1=YJ1+PI
        YJ2=YJ2+PI
        GO TO 100
 105    YJ1=YJ1-PI
        YJ2=YJ2-PI
        GO TO 100
 101    PC=0.0
        IF(YJ5*YJ6)106,106,107
 106    IF(XJ1*XJ2)108,108,111
 111    IF((YJ2-YJ1)*XJ1)125,125,126
 107    IF(XJ1*XJ2)110,105,108
 110    IF(XJ1*(Y(J1)-Y(J2))-YJ1*(X(J1)-X(J2)))125,125,126
 125    PC=-PI
        GO TO 108
 126    PC=PI
 108    CONTINUE
        CALL DAISW(16,JAIR)
        IF(JAIR.EQ.1) GO TO 39
        COL1=XC1*2.*(PC+ATAN(TAN(YJ2)/TANH(XJ2))-ATAN(TAN(YJ1)/TANH(XJ1)))
        COL2=XC2*2.*(ALOG(COSH(XJ4)-COS(YJ4))-ALOG(COSH(XJ3)-COS(YJ3)))*.5
        COL(K)=T4P*(COL1-COL2)
   39   CONTINUE
C   SPECIAL AIRFOIL EQUATIONS.
        IF(JAIR.EQ.2) GO TO 46
        COL1=4.0*XC1*2.*(PC+ATAN(YJ2/XJ2)-ATAN(YJ1/XJ1))
        COL2=2.0*XC2*2.*(ALOG(XJ4**2+YJ4**2)-ALOG(XJ3**2+YJ3**2))
        COL(K)=(T4P/4.0)*(COL1-COL2)
   46   CONTINUE
C   END SPECIAL AIRFOIL EQUATIONS.
        GOTO (22,23,21), ITEST
   22   L1=L-1
        SIGN=-1.
        IF(L1)24,24,25
   24   L1=N
        GO TO 25
   23   L1=L+1
        SIGN=1.
        IF(L1-N)25,25,27
   27   L1=1
   25   SJCL1= ((X(JC)-X(L1))**2 +(Y(JC)-Y(L1))**2 )**.5
        SLL1=((X(L)-X(L1))**2 +(Y(L)-Y(L1))**2 )**.5
        SLJC=((X(JC)-X(L))**2 +(Y(JC)-Y(L))**2 )**.5
        DSDF=T*(XP(JC)*XP(JC)+YP(JC)*YP(JC))/ADE
        DSDF=SQRT(DSDF)/(2.*PI)
        COL(K)=COL(K)+SIGN*DSDF*(1.-SJCL1/SLL1*ALOG(SJCL1/SLJC))
  21    CONTINUE
        DO 1 K=1,N2
1       Y91(K,JCO)=COL(K)
  20    CONTINUE
        WRITE(1,999)
        WRITE(1,59)
        DO 32 K=1,N2
 32     COL(K)=YP(K)
        JCO=N
        DO 2 K=1,N2
2       Y91(K,JCO)=COL(K)
        JCO=N+1
        DO 33 K=1,N2
 33     COL(K)=XP(K)
        DO 3 K=1,N2
3       Y91(K,JCO)=COL(K)
```

```
      DO 31 I=1,N2
   31 COL(I)=0.0
      JCO=N+2
      DO 4 K=1,N2
    4 Y91(K,JCO)=COL(K)
      JCO=N+3
      DO 5 K=1,N2
    5 Y91(K,JCO)=COL(K)
  999 FORMAT(1H1)
   59 FORMAT(/1X,19HFIRST APPROXIMATION/1X,14HINCOMPRESSIBLE)
      RETURN
      END
C**********************************************************************
      SUBROUTINE CFC03
      COMMON/CFCB2/ YB2(240,240)
      COMMON/CFCB3/ YB3(180,7)
      COMMON/CFCB1/ Y91(480,240)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      DIMENSION COM(480),COL(480),CO(480)
      N1=N-1
      NT=N/2+1
      NN=N+3
      DF=2.0*PI/FLOAT(N)
      YB3(1,1)=N
      YB3(2,1)=N2
      YB3(3,1)=T
      YB3(4,1)=EL
      DO 1 I=1,10
    1 YB3(I+4,1)=PROFL(I)
      YB3(15,1)=B1R
      DO 2 I=1,N2
      YB3(I,2)=X(I)
      YB3(I,3)=Y(I)
      YB3(I,4)=XP(I)
    2 YB3(I,5)=YP(I)
      DO 70 J=1,NN
      JCO=J
      DO 3 K=1,N2
    3 COM(K)=Y91(K,JCO)
      COM(N1)=1.
      IF(J-N)10,11,12
   12 IF(J-N-2)11,13,11
   13 COM(N1)=FLOAT(N)/(2.0*PI)
      GO TO 10
   11 COM(N1)=0.0
   10 DO 69 I=1,N1
      ICO=I
      DO 4 K=1,N2
    4 COL(K)=Y91(K,ICO)
      COL(N1)=1.
      CO(I)=0.0
      DO 68 K=1,N2
   68 CO(I)=CO(I)+COM(K)*COL(K)
   69 CONTINUE
      LCO=J
      DO 5 L=1,N1
    5 YB2(L,LCO)=CO(L)
   70 CONTINUE
      RETURN
      END
```

```
C**********************************************************************
      SUBROUTINE CFC04
      COMMON/CFCB2/ YB2(240,240)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      DIMENSION CO(480,480)
C  MATRIX INVERSION
      NN=N+3
      N1=N-1
      DO 40 I=1,NN
      ICO=I
      DO 1 J=1,N1
    1 CO(I,J)=YB2(J,ICO)
   40 CONTINUE
      EPS=.00000001
      DO 5 K=1,N1
      IF(CO(K,K))6,7,7
    6 U=-CO(K,K)
      GO TO 8
    7 U=CO(K,K)
    8 M=K
      IF(M-N1)9,10,10
    9 K1=K+1
      DO 11 L=K1,N1
      IF(CO(K,L))12,13,13
   12 V=-CO(K,L)
      GO TO 14
   13 V=CO(K,L)
   14 IF(U-V)15,11,11
   15 U=V
      M=L
   11 CONTINUE
      IF(U-EPS)16,16,17
   16 WRITE(1,999)
      WRITE(1,18)
      GO TO 26
   17 CONTINUE
      DO 19 J=K,NN
      B=CO(J,K)
      CO(J,K)=CO(J,M)
   19 CO(J,M)=B
   10 U=1./CO(K,K)
      DO 20 J=K,NN
   20 CO(J,K)=CO(J,K)*U
      DO 5 L=1,N1
      IF(L-K)21,5,21
   21 U=CO(K,L)
      DO 22 J=K,NN
   22 CO(J,L)=CO(J,L)-U*CO(J,K)
    5 CONTINUE
      DO 23 J=N,NN
      JCO=J
      DO 2 I=1,N1
    2 YB2(I,JCO)=CO(J,I)
   23 CONTINUE
   26 CONTINUE
   18 FORMAT(//5X,15HSINGULAR MATRIX)
   25 FORMAT(4F10.5)
  999 FORMAT(1H1)
      RETURN
      END
C**********************************************************************
```

```
      SUBROUTINE CFCOS
      COMMON/CFCH2/ YB2(240,240)
      COMMON/CFCB3/ YB3(480,7)
      COMMON/CFC99/ Y99(10,2)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      COMMON/FCFC/ XSAVE(240), CUSAVE(240)
      COMMON/BSFC/ G
      DIMENSION XP(480),YP(480),SQ(480),CU(240),X(480),SP(480),
     1CO(4,240), S(240)
      DIMENSION SS(480)
      CALL DATSW(15,N15)
      T=2.*PI/T
      N1=N-1
      NT=N/2+1
      NN=N/2
      NI=NN
      NE=NN+2
      N3=N+3
      CALL DATSW(10,J10)
      IF (J10 .EQ. 2) GOTO 202
      READ(2,103) IIRE
      N1=N1-IIRE
      NE=NE+IIRE
  202 CONTINUE
      C1X=Y99(1,2)
      C1Y=Y99(2,2)
      C2X=Y99(3,2)
      C2Y=Y99(4,2)
      DO 81 J=N,N3
      JCU=J
      JL=J-N1
      DO 15 I=1,N1
   15 CO(JL,I)=YB2(I,JCU)
      IF (N15 .EQ. 2) GOTO 303
  102 WRITE(1,106) (CO(JL,I),I=1,N1)
  303 CONTINUE
   81 CONTINUE
      DO 16 I=1,N2
      XP(I)=YB3(I,4)
   16 YP(I)=YB3(I,5)
      DO 17 I=1,N
   17 X(I)=YB3(I,7)
      IF(N15 .EQ. 2) GOTO 105
      WRITE(1,106) (XP(I),I=1,N2)
      WRITE(1,106) (YP(I),I=1,N2)
      WRITE(1,106)(X(I),I=1,N)
  105 CONTINUE
      D=SQRT(XP(NI)*XP(NI)+YP(NI)*YP(NI))
      DO 82 J=1,4
   82 SQ(J)=CO(J,NI)/D
      D=SQRT(XP(NE)*XP(NE)+YP(NE)*YP(NE))
      DO 83 J=1,4
      JF=J+4
   83 SQ(JF)=CO(J,NE-1)/D
C     THE FOLLOWING EQUATIONS APPLY TO CASCADES.
      CALL DATSW(16,JAIR)
      IF(JAIR.EQ.1) GO TO 42
      G=(C1X*(SQ(1)+SQ(5))-C1Y*(SQ(2)+SQ(6))+SQ(4)+SQ(8))/(.5*(SQ(2)+SQ(
     16))-T*(SQ(3)+SQ(7)))
   42 CONTINUE
C     THE FOLLOWING EQUATIONS APPLY TO AIRFOILS.
      IF(JAIR.EQ.2) GO TO 43
      G=(-C1X*(SQ(1)+SQ(5))+C1Y*(SQ(2)+SQ(6)))/(SQ(3)+SQ(7))
   43 CONTINUE
      CALL DATSW(12,JA)
      IF(JA .EQ. 2) GOTO 91
      READ(2,92)G
   91 CONTINUE
      DO 84 I=1,N1
      CU(I)=CO(1,I)*C1X-CO(2,I)*C1Y-G*(.5*CO(2,I)-T*CO(3,I))+CO(4,I)
      IF(JAIR.EQ.2) GO TO 45
      CU(I)=CO(1,I)*C1X-CO(2,I)*C1Y+G*CO(3,I)
   45 CONTINUE
   84 CONTINUE
      CU(N)=EL
      JB2=200
      DO 18 I=1,N
   18 YB2(I,JB2)=CU(I)
      N4=N+1
      DO 19 I=1,N
   19 SQ(I)=YB3(I,2)
      DO 21 I=N4,NN
   21 SQ(I)=YB3(I-N4+1,3)
      DO 70 K=1,N
      K2=K+N
      IF(K-NT)72,70,71
   72 JK=K
      GO TO 73
   71 JK=K-1
   73 XK=2.*PI*(-2.-SQ(K))/T
      YK=-2.*PI*SQ(K2)/T
      C1X=C1X-CU(JK)*PI/(T*FLOAT(N))*SIN(YK)/(COSH(XK)-COS(YK))
      C1Y=C1Y+CU(JK)*PI/(T*FLOAT(N))*(1.+SINH(XK)/(COSH(XK)-COS(YK)))
      XK=2.*PI*( 3.-SQ(K))/T
      C2X=C2X-CU(JK)*PI/(T*FLOAT(N))*SIN(YK)/(COSH(XK)-COS(YK))
      C2Y=C2Y+CU(JK)*PI/(T*FLOAT(N))*(1.+SINH(XK)/(COSH(XK)-COS(YK)))
   70 CONTINUE
      B1R=ATAN(C1Y/C1X)
      B1R=B1R*180./PI
      C1X=SQRT(C1X*C1X+C1Y*C1Y)
      B2R=ATAN(C2Y/C2X)
      B2R=B2R*180./PI
      C2X=SQRT(C2X*C2X+C2Y*C2Y)
      WRITE(1,999)
      WRITE(1,898) PROFL
      WRITE(1,889) T,EL,N
      WRITE(1,74)B1R,C1X,B2R,C2X
      WRITE(1,95)G
      CALL CFCOB
      DO 20 I=1,N2
   20 SQ(I)=SQRT(XP(I)*XP(I)+YP(I)*YP(I))
      DF=3.14159/FLOAT(N2)
      SP(1)=0.
      DO 11 I=2,NT
      J=I-1
      JL=J+N
      SP(I)=SP(J)+SQ(J)*DF+SQ(JL)*2*DF+SQ(I)*DF
      L=N+2-I
      JL=L+N
      J=L+1
      IF(J-N-1)10,12,12
   12 J=1
```

```
   10 SP(L)=SP(J)+SQ(J)*DF+SQ(JL)*2.*DF+SQ(L)*DF
   11 CONTINUE
      DO 1 I=1,N1
      IF(I-NT)2,3,3
    2 K=I
      GO TO 4
    3 K=I+1
    4 CONTINUE
      X(I)=X(K)
      CU(I)=CU(I)/SQ(K)
    1 CONTINUE
      DO 41 I=1,NT
      XSAVE(I)=X(I)
      CUSAVE(I)=CU(I)
   41 CONTINUE
      XSAVE(NI)=1.0
      CUSAVE(NI)=0.0
      DO 229 I=NI,N1
      XSAVE(I+1)=X(I)
      CUSAVE(I+1)=CU(I)
  229 CONTINUE
      WRITE(1,50)
      WRITE(1,49)X(1),SP(1),CU(1),X(1),SP(1),CU(1)
      DO 52 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
   52 WRITE(1,49) X(LS),SP(LS1),CU(LS),X(LP),SP(LP),CU(LP)
      READ(2,56)AM,GA
      AM1=AM
      IF(AM-.01)60,59,59
   60 DO 61 I=1,N1
   61 S(I)=CU(I)*CU(I)
      GO TO 62
59    AM=AM*SQRT((GA+1.)/(2.+(GA-1.)*AM*AM))
      WRITE(1,63)AM1,AM,GA
      SAM=GA/(GA+1.)*AM*AM*(1.-AM*AM*(GA-1.)/(GA+1.))**(1./(GA-1.))
      PGA=GA/(GA-1.)
      EGA=(GA+1.)/(GA-1.)
      DO 58 I=1,N1
      S(I)=(1.-(1.-AM*AM*CU(I)*CU(I)/PGA)**EGA)/SAM
      AML=CU(I)*AM
   58 CU(I)=AML/SQRT(1.+.5*(GA-1.)*(1.-AML*AML))
   62 WRITE(1,55)
      WRITE(1,53)X(1),SP(1),S(1),CU(1),X(1),SP(1),S(1),CU(1)
      DO 57 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
   57 WRITE(1,53)X(LS),SP(LS1),S(LS),CU(LS),X(LP),SP(LP),S(LP),CU(LP)
      SS(1)=SP(1)
      DO 291 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
      SS(LP)=SP(LP)
  291 SS(LS)=SP(LS1)
  999 FORMAT(1H1)
   51 FORMAT(3F8.3,4X,3F8.3)
   49 FORMAT(3F8.3,4X,3F8.3)
   53 FORMAT(4F8.3,4X,4F8.3)
   50 FORMAT(/1X,21HVELOCITY DISTRIBUTION/5X,12HSUCTION SIDE,15X,13HPRES
     1SURE SIDE/3X,3HX/C,5X,3HS/C,4X,5HW/L/W1,6X,3HX/C,5X,3HS/C,4X,5HW/L/W
     21/)
56    FORMAT(2F5.3)
   55 FORMAT(///1X,25H(PC1-PSL)/Q1 DISTRIBUTION/5X,12HSUCTION SIDE,15X,
     113HPRESSURE SIDE/3X,3HX/C,5X,3HS/C,4X,5HS LOC,5X,3H ML,8X,3HX/C,5X
     2,3HS/C,4X,5HS LOC,5X,3H ML/)
      WRITE(1,999)
      NC=N+1
      NH=NC/2
      CU(1)=0.
      DO 5 I=1,N
      CU(I)=ABS(CU(I))
      IF(CU(I)-4.)5,5,6
6     CU(I)=0.
5     CONTINUE
      CU(NC)=0.5
8     CU(NC)=CU(NC)+0.5
      DO 7 I=1,N
      IF(CU(NC)-CU(I))8,7,7
7     CONTINUE
      SS(NC)=1.2
      DO 9 I=1,N
      IF(SS(NC)-SP(I))13,9,9
9     CONTINUE
      GO TO 14
13    SS(NC)=1.5
14    CONTINUE
      WRITE(1,888) PROFL
      WRITE(1,889)T,EL,N
      WRITE(1,998)
      CALL PLRS(NC,CU,SS,51,61,NH)
63    FORMAT(///,1X,17HINLET MACHNUMBER=,F7.3,5X,26HCRITICAL INLET MACH N
     1UMBER=,F7.3,5X,13HGAS CONSTANT=,F7.3)
   74 FORMAT(//1X,16HINLET CONDITIONS/1X,3HB1=,F7.2,3X,6HW1/A2=,F10.5/
     11X,17HOUTLET CONDITIONS/1X,3HB2=,F7.2,3X,6HW2/W1=,F10.5/)
   92 FORMAT(F9.5)
   95 FORMAT(1X,29HNON DIMENSIONAL CIRCULATION =,F10.5)
  103 FORMAT(I2)
  106 FORMAT(1X,10F10.5)
  888 FORMAT(1X,19A4)
  889 FORMAT(1X,2HT=,F9.5,3X,3HEL=,F9.5,3X,2HN=,I3)
998   FORMAT(//1X,29HPLOT MACH NUMBER DISTRIBUTION//)
      RETURN
      END
C*********************************************************************
```

```
C*********************************************************************
      SUBROUTINE CPCOB
      COMMON/ACPC/N,N2,I,EL,PRCPL(10),BIR,PI
      COMMON/CCPC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/ECPC/ XSAVE(240), CUSAVE(240)
      COMMON/HSPC/ G
C  CALCULATE THE LIFT COEFFICIENT.
      CL=-2.0*G
C  CALCULATE THE PITCHING MOMENT.
      H=2.0*PI/FLOAT(N)
      CMPOS=0.0
      CMNEG=0.0
      DO 24 I=1,N
      CUSAVT=CUSAVE(1)
      IF(I.EQ.NT) CUSAVT=CUSAVE(I-1)
      IF(I.LT.NT.AND.X(I).LT.0.97) CMPOS=CMPOS+(1.0-CUSAVT**2)*(0.25-
     1X(I))*YP(I)*H
      IF(I.GT.NT.AND.X(I).LT.0.97) CMNEG=CMNEG+(1.0-CUSAVT**2)*(0.25-
     1X(I))*XP(I)*H
   24 CONTINUE
      CM=CMPOS-CMNEG
      WRITE(1,100) CL, CM
  100 FORMAT(//1X,'THE 2-D LIFT COEFFICIENT =',F8.5,//1X,'THE 2-D PITCHI
     1NG MOMENT COEFFICIENT =',F10.7)
      RETURN
      END
C*********************************************************************
      SUBROUTINE PLRS(N,X,Y,MX,MY,NH)
      DIMENSION X(1),Y(1),P(101),T(12)
      DATA BLANC/4H    /,STAR/4H*   /,HOR/4H-   /,VER/4HI   /
      DATA PP/4HX   /,PS/4HO   /,DUB/4H*   /
      IF(MY-101)36,36,37
37    MY=101
36    XMA=X(1)
      XMI=XMA
      YMA=Y(1)
      YMI=YMA
      DO 1 I=1,N
      XMA=(XMA+X(I)+ABS(XMA-X(I)))/2.
      XMI=(XMI+X(I)-ABS(XMI-X(I)))/2.
      YMA=(YMA+Y(I)+ABS(YMA-Y(I)))/2.
1     YMI=(YMI+Y(I)-ABS(YMI-Y(I)))/2.
      PY=(YMA-YMI)/FLOAT(MY-1)
      PX=(XMA-XMI)/FLOAT(MX-1)
      MYS=(MY-2)/10+1
      MYS=10*MYS+1
      MXS=(MX-2)/10+1
      MXS=10*MXS+1
      I=0
      DO 9 IK=1,MYS,10
      I=I+1
9     T(I)=YMI+PY*FLOAT(IK-1)+.001
      MP=1
      WRITE(1,18)(T(I),I=1,MP)
      DO 8 J=1,MXS
      TJ=FLOAT(J-1)/10.
      JJ=(J-1)/10
      IF(ABS(TJ-FLOAT(JJ))-.005)13,12,12
13    DO 5 I=1,MYS
5     P(I)=HOR
      GO TO 14
12    DO 7 I=1,MYS
7     P(I)=BLANC
14    DO 4 I=1,MYS,10
4     P(I)=VER
      DO 2 I=1,N
      MM=(XMA-X(I))/PX+1.5
      IF(J-MM)2,6,2
6     L=(Y(I)-YMI)/PY+1.5
      IND=1
      IF(I.GT.NH) IND=2
      IF(IND .EQ. 2) GOTO 52
      P(L)=PS
      GO TO 2
   52 IF(P(L)-PS)54,53,54
   53 P(L)=DUB
      GO TO 2
   54 P(L)=PP
2     CONTINUE
      XA=XMA-FLOAT(J-1)*PX
      IF(ABS(TJ-FLOAT(JJ))-.005)29,28,28
28    WRITE(1,19)(P(I),I=1,MYS)
19    FORMAT(15X,101A1)
      GOTO 8
 29   WRITE(1,31)XA,(P(I),I=1,MYS)
31    FORMAT(4X,F9.1,2X,101A1)
8     CONTINUE
      WRITE(1,18)(T(I),I=1,MP)
  18  FORMAT(11X,11(F7.2,3X))
      RETURN
      END
C*********************************************************************
      SUBROUTINE DATSW(I,J)
      COMMON/YGSW/ K(19)
      IL=I
      IF(IL.EQ.0.OR.IL.GT.18) IL=19
      J=K(IL)
      RETURN
      END
C*********************************************************************
```

```
C*****************************************************************
C*****************************************************************
C                        SFC3 MAIN PROGRAM
C*****************************************************************
C   PURPOSE
C   THIS PROGRAM CALCULATES THE SEPARATED FLOW OVER A 2-D
C   AIRFOIL.
C*****************************************************************
C   DISCRIPTION OF PARAMETERS
C           N   -NUMBER OF POINTS WHERE THE VELOCITY IS TO BE CALCULATED
C          NF   -NUMBER OF AIRFOIL POINTS READ IN
C          EL   -STAGGER ANGLE                                 (DEG.)
C           T   -PITCH TO CHORD RATIO
C       PROFL   -PROFILE DESIGNATION
C         X,Y   -TRANSFORMED COORDINATE POINT LOCATIONS ON THE PROFILE
C       XP,YP   -DX/DPHI, DY/DPHI RESPECTIVELY
C         PHI   -THE TRANSFORMED COORDINATE
C    XT,YT,YC   -THE PROFILE THICKNESS AND CAMBERLINE DISTRIBUTION
C          RC   -THE RADIUS OF CURVATURE
C          B1   -THE INLET AIR ANGLE (ANGLE OF ATTACK)        (DEG.)
C          CU   -THE VORTEX STRENGTH AND VELOCITY
C          CC   -THE COEFFICIENTS OF THE K MATRIX
C         CCL   -COLUMNS IN THE K MATRIX
C          SF   -DISTANCE ALONG THE CONTOUR
C      SDFLTS   -S/C FOR USE BY SFC4
C       CUDEL   -VELOCITY FOR USE BY SFC4
C       DELTS   -THE DISPLACEMENT THICKNESS
C           Q   -THE SOURCE STRENGTH
C           G   -CIRCULATION
C       ITMAX   -ONE LESS THAN THE MAXIMUM NUMBER OF ITERATIONS
C        TURB   -TURBULENCE LEVEL FOR B.L. CALCULATION
C        TOLL   -CONVERGENCE CRITERIA FOR ITERIATION PROCEDURE
C       REREF   -REFERENCE REYNOLDS NUMBER FOR B.L. CALCULATION
C       RMREF   -REFERENCE MACH NUMBER FOR B.L. CALCULATION
C          JI   -LOCATION WHERE KUTTA CONDITION SHOULD BE APPLIED
C*****************************************************************
C   REMARKS
C   THIS PROGRAM IS TO BE USED WITH TWO OTHER PROGRAMS (CFC1, AND CFC2)
C   ALTHOUGH IT IS A COMPLETE UNIT AS IT STANDS, IT REQUIRES THE RESULTS
C   OF CFC1 AND CFC2.
C   THE PROGRAM OPTIONS ARE GOVERNED BY DATA SWITCHES WHICH ARE READ IN
C   PROGRAM CFC1. THE OPTIONS ARE
C               1.)  DATA SWITCH 10 ON, A DIFFERENT POINT OF APPLICATION
C                                       OF THE KUTTA CONDITION IS POSSIBLE
C               2.)  DATA SWITCH 12 ON, THE CIRCULATION MAY BE READ
C                                  OFF, THE CIRCULATION IS CALCULATED
C               3.)  DATA SWITCH 13 ON, RADIUS OF CURVATURE CORRECTION
C                                  OFF, NO B.L. CURVATURE CORRECTION
C               4.)  DATA SWITCH 14 ON, INTERMEDIATE VALUES ARE PRINTED.
C                                  OFF, NO PRINTED INTERMEDIATE VALUES.
C               5.)  DATA SWITCH 15 ON, INTERMEDIATE RESULTS ARE PRINTED.
C                                  OFF, NO INTERMEDIATE RESULTS PRINTED.
C               6.)  DATA SWITCH 16 ON, THE AIRFOIL CALCULATION IS DONE
C                                  OFF, THE CASCADE CALCULATION IS DONE
C               7.)  DATA SWITCH 17 ON, BOUNDARY LAYER IS ADDED.
C                                  OFF, NO BOUNDARY LAYER.
C               8.)  DATA SWITCH 18 ON, SEPARATION IS INCLUDED.
C                                  OFF, NO SEPARATION.
C   IF THE BOUNDARY LAYER OPTION IS CHOOSEN, SFC4 SHOULD BE LINKED
C   WITH THIS PROGRAM.  SFC4 IS A SUBROUTINE CALLED BY SFC3
C*****************************************************************
      COMMON /YGSW/ ISWTCH(19)
      COMMON/CFC92/ Y82(240,210)
      COMMON/CFC91/ Y91(480,240)
      COMMON/ACFC/N,N2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/DSFC/ C1X, C1Y, C2X, C2Y
      COMMON/ESFC/ Q(480), B1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1TURB, TOLL, SSEPSS, SSEPPS, CDV(480)
      COMMON/FSFC/ XSAVE(240),CUSAVE(240)
      COMMON/A/S(32),CUBL(32),DG(32),DF(32),REREF,RMREF,TREF
      COMMON/C/CF,RCC(32),NCF,CLL(2),NK1,NDEF,IDUM
      COMMON/SW/NC,NR
      READ(3) (ISWTCH(K), K=1,19)
      READ(3) N, N2, T, EL, B1R
      READ(3) (PROFL(I), I=1,10)
      READ(3) (X(I), I=1,N2)
      READ(3) (Y(I), I=1,N2)
      READ(3) (XP(I), I=1,N2)
      READ(3) (YP(I), I=1,N2)
      READ(3) (RC(I), I=1,N2)
      READ(4) ((Y82(I,J), I=1,240), J=1,240)
      READ(4) ((Y91(I,J), I=1,480), J=1,240)
      READ(10) (XSAVE(I), I=1,N)
      READ(10) (CUSAVE(I), I=1,N)
      CALL DATSW(17,JBL)
      CALL DATSW(18,JSEP)
      PI=3.14159265
      IF(JSEP.EQ.2) GO TO 34
      IF(JBL.EQ.1) GO TO 34
      READ(1,200) SSEPSS, SSEPPS
   34 CONTINUE
      READ(1,198) B1
      READ(1,700) JI
      AM=0.0
      IF(JBL.EQ.2) GO TO 21
      READ(1,56) REREF, RMREF
      CALL DATSW(13,NC)
      READ(1,700) ITMAX
      READ(1,199) TURB
      READ(1,198) TOLL
   21 CONTINUE
      CALL SFC3
      WRITE(20) (XSAVE(I), I=1,N)
      WRITE(20) (CUSAVE(I), I=1,N)
  200 FORMAT(2E9.7)
  198 FORMAT(F7.3)
   56 FORMAT(F10.1,F10.5)
  700 FORMAT(I2)
      STOP
      END
C*****************************************************************
      SUBROUTINE SFC3
      COMMON/ESFC/ Q(480), B1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1TURB, TOLL, SSEPSS, SSEPPS, CDV(480)
      ITERBL=0
      IPRNT=2
      CALL DATSW(14,JPNT)
      CALL DATSW(17,JBL)
      IF(JPNT.EQ.1) IPRNT=1
   10 CONTINUE
      CALL SFC01
      CALL CFC03
      CALL CFC03
      CALL SFC05
      IF(IREP.EQ.1.AND.JBL.EQ.1) GO TO 10
      RETURN
      END
```

```
      SUBROUTINE SFC01
      COMMON/ACFC/N,N2,T,EL,PRCFL(10),BIR,PI
      COMMON/CFCB2/ YB2(210,240)
      COMMON/CFCB1/ Y91(480,240)
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/GSFC/ C1X, C1Y, C2X, C2Y
      COMMON/ESFC/ G(480), B1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1IOPB, IOLL, SSEPSS, SSEPPS, COV(480)
      DIMENSION SS(480),CS(480),CUX(480),CUY(480),CU(480),
     1COL(480), SO(480)
      DIMENSION CO(480)
      DIMENSION COX(480),COY(480),OF(480)
      NI1=N/2+2
      NI=N/2+1
      N1=N-1
      CALL SFC07
      QTI=0.0
      DO 57 ID=1,N
      OF(ID)= G(ID)*SQRT(XP(ID)*XP(ID)+YP(ID)*YP(ID))
   57 QTI=OF(ID)*PI/N+QTI
      DO 64 I=1,N
      CO(I)=YB2(I,240)
   64 CONTINUE
C  SHIFT THE POINTS ON THE SUCTION SIDE.
      DO 52 I=NI1,N
      J=N+NI1-I
      CO(J)=CO(J-1)
   52 CONTINUE
      CO(NI)=0.0
      DO 53 I=1,N2
      CS(I)=XP(I)/((XP(I)**2.+YP(I)**2.)**0.5)
      SS(I)=YP(I)/((XP(I)**2.+YP(I)**2.)**0.5)
      COX(I)=0
      COY(I)=0
   53 CONTINUE
      DO 54 I=1,N
      CO(I)=CO(I)/((XP(I)**2.+YP(I)**2.)**0.5)
   54 CONTINUE
      DO 55 ID=1,N
      DO 56 I=1,N2
      CF=1.
      IF(I.EQ.N2.AND.ID.EQ.1) GO TO 25
      IF(I-ID) 22,21,20
   20 IF(I-ID-N+1) 22,24,23
   23 IF(I-ID-N) 22,26,22
   24 IF(NI+1-ID) 26,21,26
   26 IF(NI-1-ID) 25,21,25
   25 CF=2.
      GO TO 21
C  THE INFLUENCE COEFFICIENTS.
   22 K=ID
      J2=K+N
      J1=K+N-1
      IF(J1-N)75,75,76

   75 J1=N2
   76 CONTINUE
      IF(JAIR.EQ.1) GO TO 93
      XCY1=-YP(K)*T/((XP(K)*XP(K)+YP(K)*YP(K))*2.*PI)
      XCX1=XP(K)*T/((XP(K)*XP(K)+YP(K)*YP(K))*2.*PI)
      XJ1=PI*(X(I)-X(J1))/T
      XJ2=PI*(X(I)-X(J2))/T
      YJ1=PI*(Y(I)-Y(J1))/T
      YJ2=PI*(Y(I)-Y(J2))/T
   93 CONTINUE
C  AIRFOIL EQUATIONS.
      IF(JAIR.EQ.2) GO TO 8
      XCY1=-YP(K)/((XP(K)*XP(K)+YP(K)*YP(K)))
      XCX1=XP(K)/((XP(K)*XP(K)+YP(K)*YP(K)))
      XJ1=(X(I)-X(J1))/2.0
      XJ2=(X(I)-X(J2))/2.0
      YJ1=(Y(I)-Y(J1))/2.0
      YJ2=(Y(I)-Y(J2))/2.0
    8 CONTINUE
      XJ3=2.*XJ1
      XJ4=2.*XJ2
  100 YJ3=2.*YJ1
      YJ4=2.*YJ2
      YJ5=ABS(YJ1)-1.570795
      YJ6=ABS(YJ2)-1.570795
      IF(YJ5)101,102,102
  102 IF(YJ6)101,101,103
  103 IF(YJ1)105,104,105
  104 YJ1=YJ1+PI
      YJ2=YJ2+PI
      GO TO 100
  105 YJ1=YJ1-PI
      YJ2=YJ2-PI
      GO TO 100
  101 PC=0.
      IF(YJ5+YJ6)106,106,107
  106 IF(XJ1*XJ2)108,109,111
  111 IF((YJ2-YJ1)*XJ1)125,125,126
  107 IF(XJ1*XJ2)110,108,108
  110 IF(XJ1*(Y(J1)-Y(J2))-YJ1*(X(J1)-X(J2)))125,125,126
  125 PC=-PI
      GO TO 108
  126 PC=PI
  108 CONTINUE
      CALL DATSW(16,JAIR)
C  EQUATIONS FOR CASCADE CALCULATION.
      IF(JAIR.EQ.1) GO TO 123
      COL1=     2.*(PC+ATAN(TAN(YJ2)/TANH(XJ2))-ATAN(TAN(YJ1)/TANH(XJ1)))
      COL2=          (ALOG(COSH(XJ4)-COS(YJ4))-ALOG(COSH(XJ3)-COS(YJ3)))
      COX(I)=COX(I)+OF(ID)/(2.*T)*(XCY1*COL1-XCX1*COL2)
      COY(I)=COY(I)+OF(ID)/(2.*T)*(XCX1*COL1+XCY1*COL2)
  123 CONTINUE
C  EQUATIONS FOR AIRFOIL CALCULATION.
      IF(JAIR.EQ.2) GO TO 121
      COL1=2.0*(PC+ATAN(YJ2/XJ2)-ATAN(YJ1/XJ1))
      COL2=(ALOG(XJ4**2+YJ4**2)-ALOG(XJ3**2+YJ3**2))
      COX(I)=COX(I)+OF(ID)/(4.0*PI)*(XCY1*COL1-XCX1*COL2)
      COY(I)=COY(I)+OF(ID)/(4.0*PI)*(XCX1*COL1+XCY1*COL2)
  121 CONTINUE
      GOTO 56
C  CALCULATE THE INFLUENCE COEFFICIENTS

   21 CN=G(ID)/(2.*CF)
      COX(I)=COX(I)-CN*SS(I)
      COY(I)=COY(I)+CN*CS(I)
   56 CONTINUE
   55 CONTINUE
C  CALCULATE THE RESULATANT VELOCITY IN THE SEPARATED ZONE.
      DO 59 ID=1,N
      K=ID
      CUX(K)=CO(K)*CS(K)          +G(ID)*SS(K)
      CUY(K)=CO(K)*SS(K)          -G(ID)*CS(K)
      CU(K)=(CUX(K)**2.+CUY(K)**2.)**0.5
   59 CONTINUE
      DO 1 I=1,N2
      XCB=XP(I)
      YCB=YP(I)
      COL(I)=YCB*COX(I)-XCB*COY(I)
    1 CONTINUE
      JCO=N+3
      DO 63 K=1,N2
      Y91(K,JCO)=COL(K)
   63 CONTINUE
    4 FORMAT(E14.7)
  201 FORMAT(F10.5,I3)
      RETURN
      END
```

```
C*********************************************************************
      SUBROUTINE SFC07
C   REMARKS
C   THIS SUBROUTINE SETS UP THE DATA FOR THE B. L. CALCULATION AND
C   CALCULATES THE STRENGTHS OF THE SOURCES FOR THE MODELLING OF THE
C   BOUNDARY LAYER AND ALSO THE SEPARATED REGION.  THERE IS SOME TRICKY
C   MANIPULATION OF ARRAYS IN ORDER TO GET THE DATA IN THE CORRECT
C   FORM FOR THE B. L. SUBPROGRAM.  THE B.L. PROGRAM REQUIRES THE
C   VELOCITY, PROFILE ARC-LENGTH, AND RADIUS OF CURVATURE.  THESE MUST
C   BE READ IN STARTING FROM THE STAGNATION POINT.  THE PROGRAM SOLVES
C   FOR THE BOUNDARY LAYER ON THE SUCTION SIDE AND THEN IN A SECOND
C   CALL FOR THE BOUNDARY LAYER ON THE PRESSURE SIDE.  THE B. L.
C   PROGRAM ALWAYS RETURNS THE LOCATION OF THE SEPARATION POINT.  IF THE
C   IS NO SEPARATION POINT, THE PROGRAM RETURNS SSEP=0.0
C   THERE IS NO LIMITATION ON WHERE SEPARATION OCCURS PROVIDING A PROPER
C   MODEL (SOURCE DISTRIBUTION) IS GIVEN.  HOWEVER THERE IS A LIMITATION
C   ON THE STAGNATION POINT LOCATION.  THE STAGNATION POINT MUST BE ON
C   THE PRESSURE SIDE OR AT THE L.E. BUT NOT ON THE SUCTION SIDE.
C*********************************************************************
      COMMON/ACFC/N,N2,T,EL,FRCFL(10),BIP,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/ESFC/ U(480), B1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1TURB, ICLL, SSEPSS, SSEPPS, CUV(480)
      COMMON/USFC/ C1X, C1Y, C2X, C2Y
      COMMON/FSFC/ XSAVE(240),CUSAVE(240)
      COMMON/GSFC/ DELTS(480), SDELTS(480), CUDEL(480), DCUDS(480), IDS,
     1IPETA(480)
      COMMON/ISFC/ CDFRI(150)
      COMMON/A/S(32),CUBL(32),DC(32),DR(32),REREF,RMREF,TREF
      COMMON/C/CF,RCC(32),NCF,CLL(2),NR1,NDEF,IDUM
      COMMON/SW/NC,NR
      DIMENSION CUXDEL(480), SP(480), SQ(480)
      NT=N/2+1
      NT1=NT+1
      NTM=NT-1
      N1=N-1
      DO 26 I=1,N
      U(I)=0.0
   26 CONTINUE
      CALL DATSW(18,JSEP)
      CALL DATSW(17,JBL)
C   DETERMINE THE STAGNATION POINT LOCATION ON THE PROFILE.
      DO 264 I=1,NT
      IF(CUSAVE(I).LT.0.0.AND.CUSAVE(I+1).GT.0.0) ISTAG=I
  264 CONTINUE
      IF(I.EQ.NT) WRITE(6,393) I
      IF(ISTAG.EQ.0) ISTAG=1
      IF(ABS(CUSAVE(ISTAG)).GT.ABS(CUSAVE(ISTAG+1))) ISTAG=ISTAG+1
C   DETERMINE THE DISTANCE ALONG THE CONTOUR.
      DO 13 I=1,N2
   13 SQ(I)=SQRT(XP(I)*XP(I)+YP(I)*YP(I))
      DF=3.14159/FLOAT(N2)
      SP(1)=0.
      DO 11 I=2,NT
      J=I-1
      JL=J+N
      SP(I)=SP(J)+SQ(J)*DF+SQ(JL)*2*DF+SQ(I)*DF
      L=N+2-I
      JL=L+N
      J=L+1
      IF(J-N-1)10,12,12
   12 J=1
   10 SP(L)=SP(J)+SQ(J)*DF+SQ(JL)*2.*DF+SQ(L)*DF
   11 CONTINUE
C   SMOOTH THE TRAILING EDGE VELOCITIES FOR THE B. L. CALCULATION.
      IF(JBL.EQ.2) GO TO 478
      DO 440 I=1,NT
      IF(X(I).GT.0.96) GO TO 441
  440 CONTINUE
  441 INP=I
      INS=2*NT-INP
      CUSLP=(CUSAVE(INP)-CUSAVE(INP-1))/(XSAVE(INP)-XSAVE(INP-1))
      CUSLS=(CUSAVE(INS+1)-CUSAVE(INS))/(XSAVE(INS+1)-XSAVE(INS))
      DO 443 I=INP,INS
      IF(I.GT.INP.AND.I.LT.NT) CUSAVE(I)=CUSAVE(INP)+CUSLP*(XSAVE(I)-
     1XSAVE(INP))
      IF(I.GE.NT.AND.I.LT.INS) CUSAVE(I)=CUSAVE(INS)+CUSLS*(XSAVE(I)-
     1XSAVE(INS))
  443 CONTINUE
C   SOLVE FOR THE BOUNDARY LAYER ON THE SUCTION SIDE.
C   SET UP THE INPUT DATA FOR THE SUCTION SIDE B.L. CALCULATION.
      IBL=0
      DO 232 I=1,ISTAG
      IBL=IBL+1
      IA=ISTAG-I+1
      RCC(IBL)=RC(IA)
      S(IBL)=SP(ISTAG)-SP(IA)
      CUBL(IBL)=ABS(CUSAVE(IA))
  232 CONTINUE
      SONE=S(IBL)
      DO 226 I=NT,N1
      IBL=IBL+1
      IA=N-I+NT
      RCC(IBL)=RC(IA)
      S(IBL)=SP(IA)+SONE
      CUBL(IBL)=ABS(CUSAVE(IA))
      IF(IBL.GE.32) GO TO 247
```

```
  220 CONTINUE
  247 NDEF=IBL
      CALL CLIBC
      SSEPSS=SSEP
      SDELTS(1)=0.0
C   DETERMINE THE SEPARATION POINT ON THE SUCTION SIDE.
  475 ISEPSS=NT
      IF(JSEP.EQ.2) GO TO 460
      IF(SSEPSS.LE.SP(N).OR.SSEPSS.GT.SP(NT)) GO TO 468
      DO 32 I=NT,N1
      IF(SP(I+1).LE.SSEPSS.AND.SP(I).GE.SSEPSS) ISEPSS=I
   32 CONTINUE
      DELX1=ABS(SP(ISEPSS)-SSEPSS)
      DELX2=ABS(SP(ISEPSS+1)-SSEPSS)
      IF(DELX2.LT.DELX1) ISEPSS=ISEPSS+1
  468 SSEPSS=SP(ISEPSS)
      WRITE(6,249) ISTAG, X(ISTAG), ISEPSS, SSEPSS
      WRITE(2,249) ISTAG, X(ISTAG), ISEPSS, SSEPSS
C   INTERPOLATE TO GET THE PROPER VALUES OF DISPLACEMENT THICKNESS AND
C   VELOCITY AT THE SOURCE LOCATIONS ON THE CONTOUR.
  460 IF(JBL.EQ.2) GO TO 482
      IDS1=IDS-1
      DO 173 I=1,IDS1
      CUXDEL(I)=CUDEL(I)*(DELTS(I+1)-DELTS(I))/(SDELTS(I+1)-SDELTS(I))
     1+DELTS(I)*DCUDS(I)
  173 CONTINUE
C   CALCULATE THE STRENGTHS OF THE SOURCES FOR THE SUCTION SIDE B. L.
      I=1
      DO 363 I=1,IDS1
      IF(SDELTS(I)-SP(ISTAG).LT.0.0.AND.SDELTS(I+1)-SP(ISTAG).GT.0.0)
     1GO TO 359
  363 CONTINUE
  359 IF(I.EQ.IDS1) WRITE(6,393) I
      IS=I+1
      JS=1
      IF(ISTAG.EQ.1) JS=1
      DO 390 I=1,ISTAG
      DO 391 J=1,JS
      IF(SDELTS(J+1).GT.SP(I).AND.SDELTS(J).LT.SP(I)) GO TO 392
  391 CONTINUE
  392 Q(I)=CUXDEL(J)+(CUXDEL(J+1)-CUXDEL(J))*(SP(I)-SDELTS(J))/
     1(SDELTS(J+1)-SDELTS(J))
      CDVIS=CDV(J)+(CDV(J+1)-CDV(J))*(SP(I)-SDELTS(J))/(SDELTS(J+1)
     1-SDELTS(J))
      CDFRI(I)=CDVIS*(XP(I)*COS(B1*PI/180.0)+YP(I)*SIN(B1*PI/180.0))
      IF(J.EQ.JS) GO TO 397
      GO TO 398
  397 IF(SP(I).LT.SDELTS(1).OR.SP(I).GT.SDELTS(IS)) WRITE(6,393)I
  398 CONTINUE
  390 CONTINUE
      DO 394 I=NT,N
      DO 395 J=JS,IDS1
      IF(SDELTS(J).LT.SP(I).AND.SDELTS(J+1).GT.SP(I)) GO TO 396
  395 CONTINUE
  396 Q(I)=CUXDEL(J)+(CUXDEL(J+1)-CUXDEL(J))*(SP(I)-SDELTS(J))/
     1(SDELTS(J+1)-SDELTS(J))
      CDVIS=CDV(J)+(CDV(J+1)-CDV(J))*(SP(I)-SDELTS(J))/(SDELTS(J+1)
     1-SDELTS(J))
      CDFRI(I)=CDVIS*(XP(I)*COS(B1*PI/180.0)+YP(I)*SIN(B1*PI/180.0))
      IF(J.EQ.IDS1) GO TO 435
      GO TO 400
  435 IF(SP(I).LT.SDELTS(JS).OR.SP(I).GT.SDELTS(IDS)) WRITE(6,393) I
  400 CONTINUE
  394 CONTINUE
C   CALCULATE THE SOURCE STRENGTHS FOR THE SEPARATED REGION ON THE S.S.
  482 IF(JSEP.EQ.2) GO TO 46
      IF(ISEPSS.LE.NT) GO TO 46
      DO 57 ID=NT,ISEPSS
      PERSEP=(SP(ID)-SSEPSS)/(SP(NT)-SSEPSS)
      VMAX=0.383
C   JACOBS DISTRIBUTION
      Q(ID)=PERSEP*VMAX/0.25
      IF(PERSEP.GT.0.25) Q(ID)=VMAX
   57 CONTINUE
   46 CONTINUE
C   SOLVE FOR THE BOUNDARY LAYER ON THE PRESSURE SIDE.
C   SET UP THE INPUT DATA FOR THE PRESSURE SIDE B.L. CALCULATION.
      IF(JBL.EQ.2) GO TO 499
      IBL=0
      DO 230 I=ISTAG,NT
      IBL=IBL+1
      IA=I
      RCC(IBL)=RC(IA)
      S(IBL)=SP(IA)-SP(ISTAG)
      CUBL(IBL)=ABS(CUSAVE(IA))
      IF(IBL.GE.32) GO TO 251
  230 CONTINUE
  251 NDEF=IBL
      CALL CLIBC
      SSEPPS=SSEP
      SDELTS(1)=0.0
C   DETERMINE THE SEPARATION POINT ON THE PRESSURE SIDE.
  499 ISEPPS=NT
      IF(JSEP.EQ.2) GO TO 491
      IF(SSEPPS.LE.SP(1).OR.SSEPPS.GT.SP(NT)) GO TO 469
      DO 47 I=1,NTM
      IF(SP(I).LE.SSEPPS.AND.SP(I+1).GE.SSEPPS) ISEPPS=I
   47 CONTINUE
      DELX1=ABS(SP(ISEPPS)-SSEPPS)
      DELX2=ABS(SP(ISEPPS+1)-SSEPPS)
      IF(DELX2.LT.DELX1) ISEPPS=ISEPPS+1
  469 SSEPPS=SP(ISEPPS)
      WRITE(6,253) ISTAG, X(ISTAG), ISEPPS, SSEPPS
      WRITE(2,253) ISTAG, X(ISTAG), ISEPPS, SSEPPS
C   INTERPOLATE TO GET THE PROPER VALUES OF DISPLACEMENT THICKNESS AND
C   VELOCITY AT THE SOURCE LOCATIONS ON THE CONTOUR.
  491 IF(JBL.EQ.2) GO TO 453
      IDS1=IDS-1
      DO 410 I=1,IDS1
      CUXDEL(I)=CUDEL(I)*(DELTS(I+1)-DELTS(I))/(SDELTS(I+1)-SDELTS(I))
     1+DELTS(I)*DCUDS(I)
  410 CONTINUE
C   CALCULATE THE STRENGTHS OF THE SOURCES FOR THE PRESSURE SIDE B. L.
      DO 430 I=ISTAG,NT
      DO 426 J=1,IDS1
      IF(SDELTS(J).LT.SP(I).AND.SDELTS(J+1).GT.SP(I)) GO TO 429
  426 CONTINUE
  429 Q(I)=CUXDEL(J)+(CUXDEL(J+1)-CUXDEL(J))*(SP(I)-SDELTS(J))/
     1(SDELTS(J+1)-SDELTS(J))
      CDVIS=CDV(J)+(CDV(J+1)-CDV(J))*(SP(I)-SDELTS(J))/(SDELTS(J+1)
     1-SDELTS(J))
      CDFRI(I)=CDVIS*(XP(I)*COS(B1*PI/180.0)+YP(I)*SIN(B1*PI/180.0))
```

```
      IF(J.EQ.IDS1) GO TO 427
      GO TO 439
  427 IF(SP(I).LT.SDELTS(1).OR.SP(I).GT.SDELTS(IDS)) WRITE(6,393) I
  438 CONTINUE
  439 CONTINUE
C   CALCULATE THE SOURCE STRENGTHS FOR THE SEPARATED REGION ON THE P.S.
  463 IF(JSEP.EQ.2) GO TO 41
      IF(ISEPPS.GE.NT) GO TO 41
      DO 43 ID=ISEPPS, NT
      PERSEP=(SP(ID)-SSEPPS)/(SP(NT)-SSEPPS)
C   SOURCE MODEL FOR AIRFOIL
      Q(ID)=0.08*PERSEP
   43 CONTINUE
   41 CONTINUE
      Q(ISTAG)=0.0
      DO 416 I=1,N
      IF(IPRNT.EQ.1) WRITE(2,413) I, ITERBL, X(I), SP(I), Q(I)
  416 CONTINUE
  413 FORMAT(1X,2I4,4F12.9)
  249 FORMAT(//1X,'THE STAGNATION POINT IS AT',I3,2X,'X=',F10.5,
     1/1X,'THE SUCTION SIDE SEPARATION POINT IS AT',I3,2X,'S/C=',F10.5)
  393 FORMAT(//1X,'INCORRECT INTERPOLATION AT POINT',1X,I3)
  253 FORMAT(//1X,'THE STAGNATION POINT IS AT',I3,2X,'X=',F10.5,
     1/1X,'THE PRESSURE SIDE SEPARATION POINT IS AT',I3,2X,'S/C=',F10.5)
      RETURN
      END
C*********************************************************************
      SUBROUTINE CFC03
      COMMON /CFC32/ Y82(240,240)
      COMMON /CFC31/ Y91(480,240)
      COMMON/ACFC/N,N2,T,EL,PRCFL(10),BIR,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      DIMENSION CON(480),COL(480),CO(480)
      N1=N-1
      NT=N/2+1
      NN=N+3
      DF=2.0*PI/FLOAT(N)
      DO 70 J=1,NN
      JCO=J
      DO 3 K=1,N2
3     CON(K)=Y91(K,JCO)
      CON(NT)=1.0
      IF(J-N)10,11,12
   12 IF(J-N-2)11,13,11
   13 CON(NT)=FLOAT(N)/(2.0*PI)
      GO TO 10
   11 CON(NT)=0.0
   10 DO 69 I=1,N1
      ICO=I
      DO 4 K=1,N2
4     COL(K)=Y91(K,ICO)
      COL(NT)=1.
      CO(I)=0.0
      DO 69 K=1,N2
   68 CO(I)=CO(I)+CON(K)*COL(K)
   69 CONTINUE
      LCO=J
      DO 5 L=1,N1
5     Y82(L,LCO)=CO(L)
   70 CONTINUE
      RETURN
      END
C*********************************************************************
      SUBROUTINE CFC04
      COMMON /CFC92/ Y82(240,240)
      COMMON/ACFC/N,N2,T,EL,PRCFL(10),BIR,PI
      COMMON/ESFC/ Q(480), B1, SSEP, JI, AM, IREP, ITERBL, IIMAX, IPRNT,
     1TURB, IOLD, SSEPSS, SSEPPS, CDV(480)
      DIMENSION CO(480,480)
C   MATRIX INVERSION
      NN=N+3
      N1=N-1
      DO 40 I=1,NN
      ICO=I
      DO 1 J=1,N1
1     CO(I,J)=Y82(J,ICO)
   40 CONTINUE
      EPS=.00000001
      DO 5 K=1,N1
      IF(CO(K,K))6,7,7
    6 U=-CO(K,K)
      GO TO 8
    7 U=CO(K,K)
    8 M=K
      IF(M-N1)9,10,10
    9 K1=K+1
      DO 11 L=K1,N1
      IF(CO(K,L))12,13,13
   12 V=-CO(K,L)
      GO TO 14
   13 V=CO(K,L)
   14 IF(U-V)15,11,11
   15 U=V
      M=L
   11 CONTINUE
      IF(U-EPS)16,16,17
   16 IF(IPRNT.EQ.1) WRITE(2,999)
      WRITE(2,18)
      GO TO 26
   17 CONTINUE
      DO 19 J=K,NN
      B=CO(J,K)
      CO(J,K)=CO(J,M)
   19 CO(J,M)=B
   10 U=1./CO(K,K)
      DO 20 J=K,NN
   20 CO(J,K)=CO(J,K)*U
      DO 5 L=1,N1
      IF(L-K)21,5,21
   21 U=CO(K,L)
      DO 22 J=K,NN
   22 CO(J,L)=CO(J,L)-U*CO(J,K)
    5 CONTINUE
      DO 23 J=N,NN
      JCO=J
      DO 2 I=1,N1
2     Y82(I,JCO)=CO(J,I)
   23 CONTINUE
   26 CONTINUE
   25 FORMAT(4F10.5)
  999 FORMAT(1H1)
   18 FORMAT(//5X,15HSINGULAR MATRIX)
      RETURN
      END
```

```
C*********************************************************************
      SUBROUTINE SFC05
      COMMON /CFCH2/ Y82(240,240)
      COMMON/ACFC/N,N2,T,EL,PRCFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/DSFC/ C1X, C1Y, C2X, C2Y
      COMMON/FSFC/ G(480), P1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1TURP, TOLL, SSEPSS, SSEPPS, CDV(480)
      COMMON/FSFC/ XSAVE(240),CUSAVE(240)
      COMMON/HSFC/ G
      DIMENSION SQ(480),CU(240),SP(480),CO(4,240)
      DIMENSION CS(480),SS(480),CUX(480),CUY(480),S(240)
      N1=N-1
      NT=N/2+1
      NH=N/2
      NI=NH
      N4=N+1
      B1R=B1*PI/180.0
      C1X=COS(B1R)
      C1Y=SIN(B1R)
      C2X=C1X
      C2Y=C1Y
      NN=N1+4
      DO 81 J=N,NN
      JCO=J
      JL=J-N1
      DO 81 I=1,N1
      CO(JL,I)=Y82(I,JCO)
   81 CONTINUE
      D=SQRT(XP(NI)*XP(NI)+YP(NI)*YP(NI))
      DO 82 J=1,4
   82 SQ(J)=CO(J,NI)/D
      D=SQRT(XP(JI)*XP(JI)+YP(JI)*YP(JI))
      DO 83 J=1,4
      JF=J+4
   83 SQ(JF)=CO(J,JI-1)/D
      CALL CAIS*(16,JAIR)
      IF(JAIR.EQ.1) GO TO 143
      G=(C1X*(SQ(1)+SQ(5))-C1Y*(SQ(6)+SQ(2))+(SQ(4)+SQ(8)))/((SQ(2)+SQ(6
     1))*.5-T*(SQ(7)+SQ(3)))
  143 CONTINUE
      IF(JAIR.EQ.2) GO TO 146
      G=(-C1X*(SQ(1)+SQ(5))+C1Y*(SQ(2)+SQ(6))-(SQ(4)+SQ(8)))/(SQ(3)+
     1SQ(7))
  146 CONTINUE
      DO 84 I=1,N1
      IF(JAIR.EQ.1) GO TO 153
      CU(I)=CO(1,I)*C1X-CO(2,I)*C1Y-G*(.5*CO(2,I)-T*CO(3,I))+CO(4,I)
  153 CONTINUE
      IF(JAIR.EQ.2) GO TO 157
      CU(I)=CO(1,I)*C1X-CO(2,I)*C1Y+G*CO(3,I)+CO(4,I)
  157 CONTINUE
   84 CONTINUE
      IF(JAIR.EQ.1) GO TO 98
      GO TO 97
   98 WRITE(2,110) B1R
      GO TO 99
   97 CONTINUE
      DO 70 K=1,N
      IF(K-NT)72,70,71
   72 JK=K
      GO TO 73
   71 JK=K-1
   73 XK=2.*PI*(-4.-X(K))/T
      YK=-2.*PI*Y(K)/T
      C1X=C1X-CU(JK)*PI/(T*N)*SIN(YK)/(COSH(XK)-COS(YK))
      C1Y=C1Y+CU(JK)*PI/(T*N)*(1.+SINH(XK)/(COSH(XK)-COS(YK)))
      XK=2.*PI*( 5.-X(K))/T
      C2X=C2X-CU(JK)*PI/(T*N)*SIN(YK)/(COSH(XK)-COS(YK))
      C2Y=C2Y+CU(JK)*PI/(T*N)*(1.+SINH(XK)/(COSH(XK)-COS(YK)))
   70 CONTINUE
      B1R=ATAN(C1Y/C1X)
      B1R=B1R*180./PI
      C1X=SQRT(C1X*C1X+C1Y*C1Y)
      B2P=ATAN(C2Y/C2X)
      B2P=B2R*180./PI
      C2X=SQRT(C2X*C2X+C2Y*C2Y)
      WRITE(2,74)B1R,C1X,B2P,C2X
   99 CONTINUE
      DO 20 I=1,N2
   20 SQ(I)=SQRT(XP(I)*XP(I)+YP(I)*YP(I))
      DF=3.14159/FLOAT(N2)
      SP(1)=0.
      DO 11 I=2,NT
      J=I-1
      JL=J+N
      SP(I)=SP(J)+SQ(J)*DF+SQ(JL)*2*DF+SQ(I)*DF
      L=N+2-I
      JL=L+N
      J=L+1
      IF(J-N-1)10,12,12
   12 J=1
   10 SP(L)=SP(J)+SQ(J)*DF+SQ(JL)*2.*DF+SQ(L)*DF
   11 CONTINUE
      DO 63 I=1,NH
      L=N+2-I
63    CU(L)=CU(L-1)
      DO 1 I=1,N
    1 CU(I)=CU(I)/SQ(I)
      DO 30 ID=1,N
      K=ID
      CS(K)=XP(K)/((XP(K)*XP(K)+YP(K)*YP(K))**0.5)
      SS(K)=YP(K)/((XP(K)*XP(K)+YP(K)*YP(K))**0.5)
      CUX(K)=CU(K)*CS(K)+G(ID)*SS(K)
      CUY(K)=CU(K)*SS(K)-G(ID)*CS(K)
   30 CU(K)=(CUX(K)*CUX(K)+CUY(K)*CUY(K))**0.5*CU(K)/ABS(CU(K))
C   CHECK FOR CONVERGENCE OF THE BOUNDARY LAYER CALCULATION.
      IREF=2
      DO 31 I=1,N
```

```
      IF(ABS((CU(I)-CUSAVE(I))/CU(I)).GT.TOLL) IREP=1
      XSAVE(I)=X(I)
      CUSAVE(I)=CU(I)
   31 CONTINUE
      CALL DATSW(17,JHL)
      IF(JHL.EQ.2) GO TO 37
      CALL DATSW(14,JPRT)
      ITERBL=ITERBL+1
      IF(ITERBL.GE.ITMAX) IREP=2
      IF(JPRT.EQ.1) GO TO 33
      IF(IREP.EQ.2) IPRNT=IPRNT-1
      IF(IPRNT.EQ.1) IREP=1
   33 IF(IREP.EQ.1) RETURN
   37 WRITE(2,36) ITERBL
      WRITE(2,701)G
      CALL SFCO8
      DO 80 I=NT,N1
80    CU(I)=CU(I+1)
      WRITE(2,50)
      WRITE(2,51)X(1),SP(1),CU(1),X(1),SP(1),CU(1)
      DO 52 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
   52 WRITE(2,51)X(LS1),SP(LS1),CU(LS),X(LP),SP(LP),CU(LP)
      IF(AM-.01)60,59,59
   60 DO 61 I=1,N1
   61 S(I)=CU(I)*CU(I)
      GO TO 62
   59 AM=SQRT(AM*AM*2.4/(2.+.4*AM*AM))
      SAM=(1.3/2.4)*AM*AM*(1.-AM*AM/6.)**2.5
      DO 58 I=1,N1
   58 S(I)=(1.-(1.-AM*AM*CU(I)*CU(I)/6.)**3.5)/SAM
   62 CONTINUE
      WRITE(2,55)
      WRITE(2,51)X(1),SP(1),S(1),X(1),SP(1),S(1)
      DO 57 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
   57 WRITE(2,51)X(LS1),SP(LS1),S(LS),X(LP),SP(LP),S(LP)
      SS(1)=SP(1)
      DO 291 LP=2,NH
      LS=N1+2-LP
      LS1=LS+1
      SS(LP)=SP(LP)
  291 SS(LS)=SP(LS1)
      WRITE(2,999)
      NC=N+1
      NH=NC/2
      CU(1)=0.
      DO 5 I=1,N
      CU(I)=ABS(CU(I))
      IF(CU(I)-1.)5,5,6
6     CU(I)=0.
 5    CONTINUE
      CU(NC)=0.5
 8    CU(NC)=CU(NC)+0.5
      DO 7 I=1,N
      IF(CU(NC)-CU(I))8,7,7
 7    CONTINUE
      SS(NC)=1.2
      DO 9 I=1,N
      IF(SS(NC)-SP(I))13,9,9
 9    CONTINUE
      GO TO 14
13    SS(NC)=1.5
14    CONTINUE
      WRITE(2,888) PROFL
      WRITE(2,889)T,EL,N
      WRITE(2,998)
      CALL PLRS(NC,CU,SS,51,61,NH)
  999 FORMAT(1H1)
  110 FORMAT(//1X,'ANGLE OF ATTACK='F8.3)
   74 FORMAT(//1X,'INLET CONDITIONS'/'B1='F7.2'     W1/W1='F10.5/1X,'OUTL
     1ET CONDITIONS    '/'B2='F7.2'     W2/W1='F10.5,/)
   36 FORMAT(//1X,'NUMBER OF ITERATIONS=',I3)
  701 FORMAT(//1X,'CIRCULATION=',F10.5)
   51 FORMAT(3F8.3,4X,3F8.3)
   50 FORMAT(/1X,'VELOCITY DISTRIBUTION'/5X'SUCTION SIDE'15X'PRESSURE SI
     1DE'/3X'X/C'5X'S/C'4X'WL/W1'8X'X/C'5X'S/C'4X'WL/W1'/)
   55 FORMAT(///'(P01-PSL)/Q1 DISTRIBUTION'/5X,'SUCTIONSIDE',15X,'PRESSU
     1RESIDE'/3X,'X/C',5X,'S/C',4X,'S LOC',8X,'X/C',5X,'S/C',4X,'S LOC'/
     2)
78     FORMAT(///,1X,17HINLET MACHNUMBER=,F7.3,5X,26HCRITICAL INLET MACHN
     1UMBER=,F7.3,5X,13HGAS CONSTANT=,F7.3)
   75 FORMAT(//1X,16HINLET CONDITIONS/1X,3HB1=,F7.2,3X,6HW1/W2=,F10.5/
     11X,17HOUTLET CONDITIONS/1X,3HB2=,F7.2,3X,6HW2/W1=,F10.5/)
   92 FORMAT(F9.5)
   95 FORMAT(1X,29HNON DIMENSIONAL CIRCULATION =,F10.5)
  193 FORMAT(I2)
  196 FORMAT(1X,10F10.5)
  888 FORMAT(1X,10A4)
  889 FORMAT(1X,2HT=,F9.5,3X,3HEL=,F9.5,3X,2HN=,I3)
  998 FORMAT(//1X,42HPLOT NON DIMENSIONAL VELOCITY DISTRIBUTION//)
      RETURN
      END
C*********************************************************************
      SUBROUTINE SFCO8
      COMMON/ACFC/N,B2,T,EL,PROFL(10),B1R,PI
      COMMON/CCFC/X(480),Y(480),XP(480),YP(480),RC(480)
      COMMON/ESFC/ G(480), B1, SSEP, JI, AM, IREP, ITERBL, ITMAX, IPRNT,
     1TURB, TOLL, SSEPSS, SSEPPS, CPV(480)
      COMMON/FSFC/ XSAVE(240),CUSAVE(240)
      COMMON/HSFC/ G
      COMMON/ISFC/ CDFRI(480)
      NT=N/2+1
      CALL DATSW(18,JSEP)
C   CALCULATE THE LIFT COEFFICIENT.
      CL=-2.0*G
C   COMPUTE THE FORM DRAG.
      H=2.0*PI/FLOAT(N)
      CXDF=0.0
      CYDF=0.0
      CMPOS=0.0
      CMNEG=0.0
      DO 10 I=1,N
      IF(JSEP.EQ.1.AND.I.EQ.NT) CUSAVT=CUSAVE(I-1)
      IF(JSEP.EQ.2.AND.I.EQ.NT) CUSAVT=0.0
      CXDF=CXDF-H*(1.0-CUSAVT**2)*YP(I)
      CYDF=CYDF+H*(1.0-CUSAVT**2)*XP(I)
      IF(I.LT.NT.AND.X(I).LT.0.97) CMPOS=CMPOS+(1.0-CUSAVT**2)*(0.25-
     1X(I))*XP(I)*H
      IF(I.GT.NT.AND.X(I).LT.0.97) CMNEG=CMNEG+(1.0-CUSAVT**2)*(0.25-
```

Copy available to DTIC does not permit fully legible reproduction

```
     1X(I))*XP(I)*H
   10 CONTINUE
      BIR=A1*PI/180.0
      CDF=CXLF*SIN(BIR)+CXDF*COS(BIR)
C  CALCULATE THE SKIN FRICTION DRAG.
      DO 25 I=1,N
      CDVISC=CDVISC+CDFRI(I)*H
      WRITE(69,194) I,X(I), XP(I), YP(I), CDVISC, CDFRI(I)
   25 CONTINUE
C  COMPUTE THE PITCHING MOMENT.
      CMPIT=(CMPOS-CMNEG)
      WRITE(2,100) CL, CDF, CDVISC, CMPIT
  100 FORMAT(//1X,'THE 2-D LIFT COEFFICIENT =',F8.5,//1X,'THE 2-D FORM
     1 DRAG COEFFICIENT=',F10.8,//1X,'THE SKIN FRICTION DRAG COEFFICIENT
     2=',F10.8,//1X,'THE 2-D PITCHING MOMENT COEFFICIENT=',F10.7)
  194 FORMAT(I3,5F12.9)
      RETURN
      END
C**********************************************************************
      SUBROUTINE PLRS(N,X,Y,MX,MY,NH)
      DIMENSION X(1),Y(1),P(101),T(12)
      DATA BLANC/4H    /,STAR/4H*   /,HOR/4H-   /,VER/4HI   /
      DATA PP/4HX   /,PS/4HO   /,DUB/4H*   /
      IF(MY-101)36,36,37
37    MY=101
36    XMA=X(1)
      XMI=XMA
      YMA=Y(1)
      YMI=YMA
      DO 1 I=1,N
      XMA=(XMA+X(I)+ABS(XMA-X(I)))/2.
      XMI=(XMI+X(I)-ABS(XMI-X(I)))/2.
      YMA=(YMA+Y(I)+ABS(YMA-Y(I)))/2.
1     YMI=(YMI+Y(I)-ABS(YMI-Y(I)))/2.
      PY=(YMA-YMI)/FLOAT(MY-1)
      PX=(XMA-XMI)/FLOAT(MX-1)
      MYS=(MY-2)/10+1
      MYS=10*MYS+1
      MXS=(MX-2)/10+1
      MXS=10*MXS+1
      I=0
      DO 9 IK=1,MYS,10
      I=I+1
9     T(I)=YMI+PY*FLOAT(IK-1)+.001
      MP=I
      WRITE(2,18)(T(I),I=1,MP)
      DO 8 J=1,MXS
      TJ=FLOAT(J-1)/10.
      JJ=(J-1)/10
      IF(ABS(TJ-FLOAT(JJ))-.005)13,12,12
13    DO 5 I=1,MYS
5     P(I)=HOR
      GO TO 14
12    DO 7 I=1,MYS
7     P(I)=BLANC
14    DO 4 I=1,MYS,10
4     P(I)=VER
      DO 2 I=1,N
      KM=(XMA-X(I))/PX+1.5
      IF(J-KM)2,6,2
6     L=(Y(I)-YMI)/PY+1.5
      IND=1
      IF(I.GT.NH) IND=2
      IF(IND .EQ. 2) GOTO 52
      P(L)=PS
      GO TO 2
   52 IF(P(L)-PS)54,53,54
   53 P(L)=DUB
      GO TO 2
   54 P(L)=PP
2     CONTINUE
      XA=XMA-FLOAT(J-1)*PX
      IF(ABS(TJ-FLOAT(JJ))-.005)29,28,28
28    WRITE(2,19)(P(I),I=1,MYS)
19    FORMAT(15X,101A1)
      GOTO 8
 29   WRITE(2,31)XA,(P(I),I=1,MYS)
31    FORMAT(4X,F9.1,2X,101A1)
8     CONTINUE
      WRITE(2,18)(T(I),I=1,MP)
  18  FORMAT(11X,11(F7.2,3X))
      RETURN
      END
C**********************************************************************
      SUBROUTINE DATSW(I,J)
      COMMON/YGSW/ K(19)
      IL=I
      IF(IL.EQ.0.OR.IL.GT.18) IL=19
      J=K(IL)
      RETURN
      END
C**********************************************************************
```

SFC.4 FOR

```
C       PROGRAM CLIBC
      SUBROUTINE CLTBC
      COMMON/A/S(32),Q(32),DO(32),DP(32),REREF,RMREF,TREF
      COMMON/B/PRMI(5),Y(2),DERY(2),NDIM,IHLF
      COMMON/C/CF,RC(32),NCF,CLL(2),NP1,NDEF
      COMMON/GAZ/W,GA,R
      COMMON/SW/NC,NR
      COMMON/R/RE1
      COMMON/SAVE/ TEST1, TEST2, ITEST
      COMMON/ESFC/ DUMM(481), SSEP, DUM(5), IPRNT, TURB, TOLL, DUMI(2),
     1COV(480)
      COMMON/GSFC/ DELTS(480), SDELTS(480), CUDEL(480), DCUDS(480), IDS,
     1THETA(480)
      COMMON/TABLE/TAB1(21),TAB2(21),TAB3(21),TAB4(21),TAB5(21),TAB6(21)
     1,TAB7(21),TAB8(21),TAB9(21),TAB10(21),TAB11(21),TAB12(21),TAB13(21
     2),TAB14(21),TAB15(21),TAB16(21)
      DIMENSION RA(32),SS(32),CBSL(32)
      DIMENSION TITRE(15)
C*************************
      IDS=0
      DO 874 I=1,480
      COV(I)=0.0
  874 CONTINUE
      SSEP=0.0
      ITEST=0
      NP1=0
      NCF=0
      IHLF=0
      RE1=0.0
      PRMI(5)=0.0
      DO 10 I=1,2
      CLL(I)=0.0
      Y(I)=0.0
   10 CONTINUE
      TEST1=0.0001
      TEST2=0.01
C*************************
C       READ(1,3)TITRE
   3  FORMAT(15A4)
      IF(IPRNT.EQ.1) WRITE(2,2) TITRE
   2   FORMAT(///,1X,15A4)
      IF(IPRNT.EQ.1) WRITE(2,95)
  95  FORMAT(1X,41HCALCULATION OF LAMINAR AND TURBULENT B.L.,/
     11X,30HMFIELD OF LEFCLL-PAPAILICU-HUO,//)
C       READ(1,901) NDEF
 901  FORMAT (I3)
C
C       CONDITIONS FOR AIR
C
C       READ(1,107)W,GA,R
      W=0.75
      GA=1.4
      R=[illegible]
 107  FORMAT(3F6.1)
C       READ(1,104) REREF,RMREF
      RMH=RMREF
C  MODIF F 2 78        CARTES AVANCEES
      RMREF=RMH**2*(GA+1.)/(2.+(GA-1.)*RMH**2)
      RMC   =SQRT(RMREF)
C
      TREF=1./(1.+(GA-1.)*RMH**2/2.)
 104  FORMAT(F10.0,F7.5)
C       READ(1,105) TURB
C       READ(1,105) (S(I),Q(I),RC(I),RA(I),CBSL(I),I=1,NDEF)
 105  FORMAT(5F10.5)
C       READ(1,108)NC
C       READ(1,108)NR
C       READ(1,108)NT
C        READ(1,108)ND
C****************************************
      NR=2
      NT=1
      ND=2
C****************************************
      IF(IPRNT.EQ.1.AND.NC.EQ.1) WRITE (2,125)
 125  FORMAT(1H0,'CURVATURE CORRECTION IS DONE')
      IF (IPRNT.EQ.1.AND.NR.EQ.1) WRITE (2,126)
 126  FORMAT(1H0,'ROTATION CORRECTION IS DONE',///)
 108  FORMAT(I1)
C
C   IF NC=1,CURVATURE CORRECTION FOR CD IS REQUIRED
C   IF NC =2,NO CURVATURE CORRECTION IS DONE
C   IF NR=1,ROTATION CORRECTION FOR CD IS REQUIRED
C   IF NR=2,NO ROTATION CORRECTION IS DONE
C IF NT=1 CALCULATION STARTS IN LAMINAR REGION
C IF NT=2 ONLY TURBULENT BOUNDARY LAYER IS CALCULATED
C WHEN NT=2,STARTING VALUES FOR TURBULENT BOUNDARY LAYER MUST BE INPUT
C IF ND=1,DIVERGENCE     DS=SQRT(DX**2+DR**2)
C IF ND=2,NO DIVERGANCE TWO DIMENSIONAL
C
      GO TO (775,776),NR
 775  CONTINUE
      READ(1,129)RPM
 129  FORMAT(F10.4)
      READ(1,131) SIGN
      RPM=RPM*SIGN
 131  FORMAT(F4.1)
 776  CONTINUE
C
C        SIGN=1. POUR UNE FACE MENEE
C        SIGN=-1. POUR UNE FACE MENANTE
C
      GO TO (98,97),NC
  97  DO 99 I=1,NDEF
  99  RC(I)=0.
  98  CONTINUE
      GO TO (58,57),NR
  58  U2=RPM*2.*3.1416/60.
      DO 56 I=1,NDEF
      RC(I)=RC(I)*Q(I)/(Q(I)+U2*CBSL(I)*RC(I))
      IF (RC(I).EQ.0) RC(I)=Q(I)/(CBSL(I)*U2)
      IF(NC.EQ.2)RC(I)=Q(I)/(CBSL(I)*U2)
  56  CONTINUE
  57  CONTINUE
      GO TO (136,137),ND
 137  DO 138 I=1,NDEF
 138  DR(I)=0
      GO TO 774
 136  CONTINUE
      DO 511 I=1,NDEF
 511  SS(I)=S(I)
      DO 53 I=2,NDEF
```

```
   53  S(I)=S(I-1)+SQRT((SS(I)-SS(I-1))**2+(RA(I)-RA(I-1))**2)
C
C      FOR THE AXISYMETRIC CASE (WITH RADIUS GIVEN),
C      S-INPUT ARE THE LENGHT IN AXIAL DIRECTION
C      WE TRANSFORM THEM TO THE REAL ARC LENGTH (ALONG THE SURFACE)
C
       CALL LENIR(NDEF,S,RA,DR)
       DO 74 I=1,NDEF
       IF(DR(I).EQ.0.) GO TO 74
       IF (RA(I).EQ.0.)GO TO 74
       DR(I)=DR(I)/RA(I)
   74  CONTINUE
  774  CONTINUE
       PRMT(1)=S(1)
       PRMT(2)=S(NDEF)
       IF(IPRNT.EQ.1) WRITE(2,109)
  109  FORMAT(5X,1HS,9X,1HU,8X,2HRC,8X,2HRA,8X,4HCBSL,/)
       IF(IPRNT.EQ.1) WRITE(2,106) (S(J),U(J),RC(J),RA(J),CBSL(J),J=1,
      1NDEF)
       IF(IPRNT.EQ.1) WRITE(2,106) PRMT(1),PRMT(2)
  106  FORMAT(1X,5F10.5)
C   MODIF DU 8 2 79
C      PRMT(3)=0.0005
C      MODIF 12-2-79
C      PRMT(3)=0.005
       PRMT(3)=0.005*PRMT(2)
C
       PRMT(4)=0.002
       DERY(1)=.99
       DERY(2)=.01
       NDIM=2
       GO TO (132,133),NT
C
C      ONLY TURBULENT B.L. CALCULATION
C
  133  READ(1,134)NSTS,U,H12,D2
C MODIF DU 7/2/78
       RMREF=RMC
       RMS=U*RMREF
       RM2=(RMS*RMS/(1.+(GA-1.)/2.*(1.-RMS*RMS)))
       H12I=(H12-.272H*RM2)/(1.+0.1145*RM2)
       H32I=FH32(H12I)
       RM=SQRT(RM2)
       H32=H32I*(1.+0.007333*RM)
C      H32=FH32(H12)
C
       D3=D2*H32
       STS=S(NSTS)
       CBL(1)=D2*H12
       CBL(2)=D2
  134  FORMAT(I2,3F10.5)
       IDIR=2
       GO TO 135
  132  CONTINUE
C
C      LAMINAR AND TURBULENT CALCULATION
C
       IF(IPRNT.EQ.1) WRITE(2,101)
  101  FORMAT(1H ,' EXEC CBLDLE')
       RMREF=RMH
       CALL HRLST(H32,D3,U,TURM,IDIR,STS)
       RMREF=RMC
       IF(IPRNT.EQ.1) WRITE(2,103)
  103  FORMAT(/' PHASE IS EXECUTED')
C
C      STS IS TRANSITION OR SEPARATION POINT
C
  135  CONTINUE
       GO TO (51,52,51),IDIR
   52  CONTINUE
       RMS=U*RMREF
       RM=SQRT(RMS*RMS/(1.+(GA-1.)/2.*(1.-RMS*RMS)))
       RE3=RERREF*U*D3/(TREF*(1.+(GA-1.)*RM**2/2.))**ABS(1.+W-GA/(GA-1.))
       IF(H32-1.5150)201,201,202
  201  H32I=1.515
       H12I=2.407958
       H13I=1.569411
       GO TO 203
  202  H32I=H32/(1.+.011*RM/1.5)
       A=1.
       B=2.954101
   15  Z=(A+B)/2.
       H=0.4422849E01-0.4598586E01*Z+0.3287949E01*Z**2-0.1227710E01*Z**3+
      10.2380595*Z**4-0.1876198E-1*Z**5
       DF=H-H32I
       IF(ABS(DF)-.00005 )11,11,12
   12  IF(DF)13,13,14
   13  B=Z
       GO TO 15
   14  A=Z
       GO TO 15
C
C
   11  H12I=Z
       H13I=H12I/H32I
  203  CONTINUE
       RL=FL(H13I,RM)
       AFA=(R- W )*(GA-1.)*RM**2/2.
       X=ALOG(RE3)+2.*(1.+AFA)*RL
       RE1=RE3*H13I
C   H12I IS FOUND
       PRMT(1)=STS
C
C      RL IS VALUE OF L AT TRANSITION POINT (TBL)
C   X IS VALUE OF X AT TRANSITION POINT (TBL)
C
       IF(IPRNT.EQ.1) WRITE(2,114)RL,X
  114  FORMAT(1H1,'INITIAL VALUES FOR TURB. B. L.'/' L=',F10.5,'  X=',
      AF10.5)
C      CALCULATION OF T.B.L.
       Y(1)=RL
       Y(2)=X
       IF(IPRNT.EQ.1) WRITE(2,102)
  102  FORMAT(/' EXEC CBLCTWO')
       DO 304 I=2,NDEF
       IL=I-1
       S(IL)=S(I)
       RC(IL)=RC(I)
       RA(IL)=RA(I)
  304  U(IL)=U(I)
       NDEF=NDEF-1
       CALL HUOLF
```

```
      IF(IPRNT.EQ.1) WRITE(2,103)
51    CONTINUE
      RETURN
      END
      BLOCK DATA
      COMMON/TABLE/TAB1(21),TAB2(21),TAB3(21),TAB4(21),TAB5(21),TAB6(21)
     1,TAB7(21),TAB8(21),TAB9(21),TAB10(21),TAB11(21),TAB12(21),TAB13(21
     2),TAB14(21),TAB15(21),TAB16(21)
      DATA TAB1 /      .47455191E+0, .11604726E+2,-.11027681E+4,
     1 .35814069E+5,-.69938285E+5, .68170101E+7,-.25195278E+8,
     2 .16216109E+0, .11517770E+2,-.10664505E+4, .34259533E+5,
     3-.60913181E+6, .65556861E+7,-.24605546E+8, .42682689E+9,
     4 .11935620E+2,-.10481637E+4, .33753167E+5,-.67045713E+6,
     5 .60840830E+7,-.25601822E+8/
      DATA TAB2 /      .37722993E+0, .12717128E+2,-.10857642E+4,
     1 .36561673E+5,-.76259857E+6, .79507400E+7,-.31817294E+8,
     2 .51504184E+0, .15115697E+2,-.50601539E+3, .75613595E+4,
     3-.70581001E+5, .33441011E+6,-.60950132E+6, .49984698E+0,
     4 .15236330E+2,-.53564724E+3, .76337971E+4,-.72922698E+5,
     5 .35360629E+6,-.65935641E+6/
      DATA TAB3 /      .15453214E+0, .17102795E+2,-.60689431E+3,
     1 .10876292E+5,-.11693699E+5, .61887687E+6,-.12450802E+7,
     2 .40030563E+0, .16217226E+2,-.60189379E+3, .11599595E+5,
     3-.13700015E+5, .79559475E+6,-.17576926E+7, .53116610E+0,
     4 .16138170E+2,-.28537161E+3, .27745604E+4,-.17849941E+5,
     5 .58429559E+5,-.72913623E+5/
      DATA TAB4 /      .51479793E+0, .16024417E+2,-.24981238E+3,
     1 .24025119E+4,-.19201824E+5, .65111243E+5,-.83703935E+5,
     2 .56839959E+0, .16115571E+2,-.31907546E+3, .35452921E+4,
     3-.25697396E+5, .53459248E+5,-.12934583E+6, .40840905E+0,
     4 .15956279E+2,-.35627175E+3, .45747561E+4,-.37902433E+5,
     5 .15567151E+6,-.24273181E+6/
      DATA TAB5 /      .53798366E+0, .15501667E+2,-.18537945E+3,
     1 .13423254E+4,-.68571413E+4, .17627319E+5,-.17479961E+5,
     2 .52165521E+0, .15446082E+2,-.19251430E+3, .14744431E+4,
     3-.73762613E+4, .21236569E+5,-.21554983E+5, .47374713E+0,
     4 .15629759E+2,-.22106875E+3, .19721389E+4,-.11764775E+5,
     5 .34559451E+5,-.38734751E+5/
      DATA TAB6 /      .41291178E+0, .15627599E+2,-.26294750E+3,
     1 .24295526E+4,-.19563908E+5, .65915060E+5,-.83364123E+5,
     2 .55950320E+0, .13458675E+2,-.10317117E+3, .51850754E+3,
     3-.23110253E+4, .53112658E+4,-.45222152E+4, .53941100E+0,
     4 .13679933E+2,-.11352933E+3, .63494266E+3,-.29734069E+4,
     5 .70601191E+4,-.61621917E+4/
      DATA TAB7 /      .39217206E+0, .13614040E+2,-.12922298E+3,
     1 .66048111E+3,-.45942343E+4, .11977309E+5,-.11460745E+5,
     2 .42936591E+0, .13511568E+2,-.15450274E+3, .12995059E+4,
     3-.60797311E+4, .23995298E+5,-.26176751E+5, .58399630E+0,
     4 .11324013E+2,-.33376283E+2,-.13036670E+3, .62965554E+3,
     5-.67999305E+3, .42519497E+3/
      DATA TAB8 /      .56220659E+0, .11513973E+2,-.39177126E+2,
     1-.10090256E+3, .59338809E+3,-.63460444E+3, .24992172E+3,
     2 .51244531E+0, .11452602E+2,-.46817590E+2,-.51591199E+2,
     3 .27743571E+3,-.81097363E+2,-.26936377E+3, .44690157E+0,
     4 .11310621E+2,-.59635770E+2, .55245605E+2,-.36329615E+3,
     5 .17395419E+1,-.21531610E+4/
      DATA TAB9 /      .25175982E-1,-.35778088E-1, .17623812E-1,
     1-.77206817E+0, .15393014E+1,-.50353205E+1, .76937562E+1,
     2 .28590539E-1,-.33503109E-1, .23156187E-1,-.86415487E+0,
     3 .20223159E+1,-.52605450E+1, .67601483E+1, .22831334E-1,
     4-.26933610E-1, .13009892E-1,-.76830476E+0, .13736563E+1,
     5-.43466932E+1, .62224733E+1/
      DATA TAB10/      .26379829E-1,-.19668437E-1, .11164811E-1,
     1-.72351282E+0, .10423011E+1,-.36159837E+1, .45993341E+1,
     2 .24692986E-1,-.31728577E-1, .57506626E-1,-.51901231E+0,
     3 .10421168E+1,-.10268111E+1, .11394012E+1, .23975992E-1,
     4-.30090669E-1, .52179342E-1,-.56244946E+0, .10071184E+1,
     5-.15800083E+1, .10938818E+1/
      DATA TAB11/      .21961153E-1,-.26091002E-1, .44230652E-1,
     1-.50090396E+0, .11224813E+1,-.19574074E+1, .11299919E+1,
     2 .19290561E-1,-.21801227E-1, .42046857E-1,-.54148525E+0,
     3 .14447486E+1,-.29596776E+1, .22443092E+1, .24430396E-1,
     4-.32707831E-1, .69701255E-1,-.45674104E+0, .10214282E+1,
     5-.12958782E+1, .61012381E+0/
      DATA TAB12/      .23566530E-1,-.31423556E-1, .65459330E-1,
     1-.45644265E+0, .10777223E+1,-.11576769E+1, .73571413E+0,
     2 .21491771E-1,-.27960467E-1, .52348100E-1,-.42196852E+0,
     3 .10929442E+1,-.16761572E+1, .94813266E+0, .19752014E-1,
     4-.21163032E-1, .41082346E-1,-.39447075E+0, .11466525E+1,
     5-.20467461E+1, .13003457E+1/
      DATA TAB13/      .24113345E-1,-.33807194E-1, .72879851E-1,
     1-.42681796E+0, .10284441E+1,-.12723428E+1, .56782465E+0,
     2 .23351312E-1,-.32568723E-1, .66200495E-1,-.40568695E+0,
     3 .10073469E+1,-.12963697E+1, .60198230E+0, .21220017E-1,
     4-.29310720E-1, .51190877E-1,-.35692691E+0, .96876519E+0,
     5-.13979518E+1, .72358013E+0/
      DATA TAB14/      .18451023E-1,-.25904590E-1, .38066466E-1,
     1-.31408906E+0, .95890755E+0,-.16099035E+1, .94729951E+0,
     2 .23982411E-1,-.34931577E-1, .74957556E-1,-.40704263E+0,
     3 .99858176E+0,-.11996049E+1, .51724512E+0, .23206788E-1,
     4-.33651596E-1, .66887457E-1,-.37902076E+0, .96128273E+0,
     5-.12011480E+1, .53793544E+0/
      DATA TAB15/      .21051032E-1,-.30471633E-1, .44268026E-1,
     1-.31302196E+0, .56693156E+0,-.12054426E+1, .59534171E+0,
     2 .16260186E-1,-.26959344E-1, .32971203E-1,-.24906639E+0,
     3 .79615452E+0,-.12566039E+1, .73031217E+0, .77566719E-2,
     4-.42021933E-1, .13135028E-1, .10662634E+0,-.55953020E-1,
     5-.12928462E+0, .10709644E+0/
      DATA TAB16/      .19332659E-2,-.41116765E-1, .12574119E-1,
     1 .13417697E+0,-.11668355E+0,-.13034249E-1,-.38387757E-1,
     2 .12064281E-1, .82369805E-1,-.66963613E-1, .22925326E+0,
     3-.30147624E+0, .14953393E+0,-.27349022E-1,-.35244142E-1,
     4 .10994678E-1,-.17775149E-1, .29233402E+0,-.54570251E+0,
     5 .49452872E+0,-.21098114E+0/
      END
C     SUBROUTINE SBTCT(RM,RL,RX,FFBT,FFCT,RCA,GA)
      SUBROUTINE SBTCT(RM,RL,RX,FFBT,FFCT,RCA,GA,DGA)
C
C SUBROUTINE TO CALCULATE THE COEFFICIENTS B1 AND CT
C FOR COMPRESSIBLE TURBULENT BOUNDARY LAYERS
C THE CURVATURE CORRECTION FOR CD IS INCLUDED
C RCA IS FOUND FROM FO
C FO IS FOUND FROM LI,XI
C CF IS FOUND THEN WITH THIS RCA       AND WITH THE SUBROUTINE SF
C VERSION C OF SBTCT 18. 8. 72
C
      COMMON/A/S(32),U(32),DG(32),DR(32),RERef,RMREF,IREF
      COMMON/C/CF,RC(32),NCF,CLL(2),RR1,LNI
      COMMON/GAZ/W,GA,R
      COMMON/SG/BC,NR
      COMMON/F/RE1
      RMY=R*(GA-1.)*RM*RM/2.
```

```
      AA=.5*(R-W)*(GA-1.)*RM*RM
C
C COD IS FOUND FROM FFCTO
C
      IF(RL)50,50,51
   50 RL=.0002
   51 Z=SQRT(RL)
      FFCTO=FRG(2,RM,Z,RX)
      RMTO=ERG(1,RM,RL,RX)
      FFBTO=-RMTO/(1.+2.*(1.+AA)*RMTO)
      CDO=FFCTO/((1.+2.*(1.+AA)*RMTO)*EXP(2.*(1.+AA)*RL))
      CALL SFH13(RM,RL,H131,H121,H13)
      H32=H121/H131*(1.+0.011*RM/1.5)
      H12=FH(H121,RM)
C
C CF IS FOUND FROM RMTO AND CDO, H32
C
      CF=2.*CDO/H32*(1.-RMTO*(H12-1.-2.*RMM))
C
C CALCULATION OF CD WITH CURVATURE CORRECTION
C
C
C
C RLI IS THE INCOMPRESSIBLE VALUE OF RL CORRESPONDING TO THE SAME H13I
C
      RMI=.001
      RLI=FL(H131,RMI)
C
C RXI IS THE CORRESPONDING VALUE OF RX CORRESPONDING TO RLI AND THE SAME
C RE-NO RE1
C
      RE1=EXP(RX-2.*(1.+AA)*RL)*H13
      RXI=ALOG(RE1/H131)+2.*RLI
C
C  RLA IS FOUND BY ITERATING IN FOLLOWING SYSTEM
C  FU =F(RLA,RE1)
C  G  =F(RLA)
C  FOR A GIVEN H12 VALUE
C  H12=F(G,FU)
C  THESIS EQ. (2-17),(2-24),(2-35)
C
      CALL HRA(H121,DUM,RLA)
      A=ALOG(RE1)
      B=ALOG(1.626+1.2195*RLA)
      FU=.41/(RLA+1.55+A-B)
C
C SDEL GIVES THE RATIO DEL*DELTA1/DELTA
C
      CA=SQRT(RMM/(1.+RMM))
      CB=ATAN(CA/SQRT(1.-CA*CA))
      TETW=1./(1.+RMM)
      RNU=TETW**(1.+W)
      CALL SF(RMM,RLA,FU,RE1,CA,CB,RNU,DEL,F)
      CF=2.*F*F*TETW
      IF (NCF-1) 71,71,72
   71 NCF=2
      GO TO 2
   72 CONTINUE
C
C IF NC#2, NO CURVATURE CORRECTION FOR CD IS DONE
C IF NC#1, CURVATURE CORRECTION FOR CD IS REQUIRED
C RC IS THE RADIUS OF CURVATURE/SREF, ONLY NEEDED IF NC#2
C
C
C  IF NR=2 NO ROTATION CORRECTION FOR CD IS DONE
C  IF NR=1 ROTATION CORRECTION FOR CD IS REQUIRED
C
      IF((NR.EQ.2).AND.(NC.EQ.2)) GO TO 2
C
C RC POSITIVE, SUCTION SURFACE$CONVEXED<, BB#7.
C RC NEGATIVE, PRESSURE SURFACE$CONCAVED<, BB#4.
C
      CALL ONE(GA,GA,RMREF,RMS,RM,UA)
      TA=1./FT(GA,RM)
      TA=(TA/TREF)**(1.+W-GA/(GA-1.))
C
C DELS#DELTA1/SREF
C DELR#DELTA/CURVATURE RADIUS
C
      DELS=RE1/REREF*TA/UA
      DELR=DELS/(DEL*RCA)
C
C     MODIF DU 7 6 78
C
      DELRT=-.3
      IF(DELR.LT.DELRT)DELR=DELRT
      IF (RCA-0.) 3,3,4
    3 BB=4.
C     CORRECTION OF CD FOR UNSTABLE SIDE
C
      GO TO 15
    4 BB=7.
C
C     CORRECTION FOR STABLE SIDE
C     VKI IN.37  PAPAILIOU,NURZIA & SATTA
C
C SAB GIVES THE COEFFICIENTS A,B FOR THE CD-CORRECTION
C
  15  CONTINUE
      CALL SAB(RM,H121,A,B)
      DCD=8.*BB*DELR*(-A+BB*DELR*B)
      CDC=CDO+DCD
      IF(RL.LT..25) GO TO 5
      CDC=CDC+((.32-RL)/.07)**3 *DCD
    5 CONTINUE
C
      IF(CDC.LT.0.) WRITE(2,16)
  16  FORMAT(1X,'END OF CALCULATION    CD<0. ')
C     END OF MODIF 7 6 78
C
      RMT=(1.-CF*H32/(2.*CDC))/(H12-1.-R*(GA-1.)*RM**2)
      FFBT=-RMT/(1.+2.*(1.+AA)*RMT)
      FFCT=CDC*(1.+2.*(1.+AA)*RMT)*EXP(2.*(1.+AA)*RL)
      GO TO 13
  2   FFBT=FFBTO
      FFCT=FFCTO
   13 RETURN
      END
C** TITRE DRUCK
      SUBROUTINE DRUCK(SA,Y,DERY,IHLF,NDIM,PRMT,CLL)
C
C OUTPUT SUBROUTINE FOR THE MAIN PROGRAM DIR31 AND SUBROUTINE ENGS
C DATE 26. 2 72
```

```
C
      COMMON/A/S(32),Q(32),DQ(32),DR(32),REREF,RMREF,TREF
C MODIF DU 8 2 78
C     COMMON/C/CF,RC(32),NCF,DUM(3)
      COMMON/C/CF,RC(32),NCF,DUM(2),NDUM,NDEF
      COMMON/GSFC/ DELTS(480), SDELTS(480), CUDEL(480), DCUDS(480), IDS,
     1THETA(480)
C
      COMMON/GAZ/W,GA,R
      COMMON/SAVE/ TEST1, TEST2, ITEST
      COMMON/ESFC/ DUMMO(481), SSEP, DUMR(5), IPRNT, DUMT(4), CDV(480)
      DIMENSION Y(2),DERY(2),PRMT(5),CLL(2)
C  MODIF DU 8 2 78
C*******************************
C        DATA ITEST/0/
C*******************************
C
C
C
C CHECK IF SEPARATION OCCURS
C
      IF(Y(1)-PRMT(4))1,1,2
C
C SEPARATION POINT IS REACHED
C
    1 PRMT(5)=1.
C**************************
      SSEP=SA
C**************************
      WRITE(2,11)SA
      WRITE(6,11) SA
   11 FORMAT(//' SEPARATION AT S/SREF=',F5.2)
      RETURN
C
C SUBROUTINE USED TO CONVERT Q INTO MACH NO. RM AND LAVAL NO. RMS
C IF THE REFERENCE LAVAL NO. RMS1 IS GIVEN
C DATE 22. 1. 72
C
C MODIF DU 7/2/78
    2 IF(INDEF.GT. 10) GO TO 16
      IF(ITEST)13,13,14
   13 DO 10 I=1,NDEF
      ITEST=I-1
      IF(SA-S(I))12,12,10
   10 CONTINUE
   14 IF(SA-S(ITEST))15,15,12
   15 RETURN
   12 ITEST=ITEST+1
   16 CONTINUE
C   2 CALL QMF(QA,GA,RMREF,RMS,RM,UA)
      CALL DQDSD(SA,QA,DQA,DRA,RCA)
      CALL QMF(QA,GA,RMREF,RMS,RM,UA)
C
      CALL SFH13(RM,Y(1),H13I,H12I,H13)
      AFA=(R-W)*RM**2/2.*(GA-1.)
      RE3=REF*(Y(2)-2.*(1.+AFA)*Y(1))
      TA=1./F1(GA,RM)
      TA=TREF/TA**(1.+W-GA/(GA-1.))
      DEL3=RE3/(REREF*UA)/TA
      H32=H12I/H13I*(1.+0.011*RM/1.5)
      H12=H12I*(1.+RM**2*.1145)+1.55*.176*RM**2
      DEL2=DEL3/H32
      DEL1=DEL3*H13
      RE2=RE3/H32
C***********************
      IF(SA.EQ.SDELTS(IDS)) GO TO 222
      IDS=IDS+1
      DELTS(IDS)=DEL1
      CUDEL(IDS)=UA
      SDELTS(IDS)=SA
      DCUDS(IDS)=DQA
      THETA(IDS)=DEL2
C COMPUTE THE SKIN FRICTION DRAG.
      IF(IDS.EQ.0) SASAV=0.0
      CDV(IDS)=+UA**2*CF
  222 CONTINUE
C***********************
      IF(IPRNT.EQ.1) WRITE(2,3)SA,UA,DQA,Y(1),Y(2),H12I,H12,RE2,DEL1,
     1DEL2,DEL3
    3 FORMAT(2X,7F10.4,4E14.6)
      CLL(1)=DEL1
      CLL(2)=DEL2
      DUM(1)=DEL1
      DUM(2)=DEL2
      RETURN
      END
C** TITRE DQDSD   (VERSION CDLCX)
      SUBROUTINE DQDSD(SA,QA,DQA,DRA,RCA)
C
C INTERPOLATION TO FIND Q AND DQ/DS
C SA=ARC LENGTH/SREF
C QA=QN(Q/UREF) AT SA
C DQA=DQ/DS AT SA
C DRA=DR/DS/R OR 1/(S-S0)
C DATE 7. 3. 72
C
      COMMON/A/S(32),Q(32),DQ(32),DR(32),REREF,RMREF,TREF
      COMMON/C/CF,RC(32),NCF,DUM(3)
      DIMENSION Z(4),V(4)
C  MODIF DU 8 2 78
C     NP=3
      NP=2
C
      I=NP
    9 IF (SA-S(I)) 7,6,8
    6 QA=Q(I)
      DQA=DQ(I)
      DRA=DR(I)
      RCA=RC(I)
      GO TO 12
    8 I=I+1
C  MODIF DU 8 2 78
      NP=3
C
      GO TO 9
    7 DO 10 J=1,NP
      II=I-NP+J
      Z(J)=S(II)
   10 V(J)=Q(II)
      N=NP
      CALL LARAN(Z,V,N,SA,QA)
      DO 13 J=1,NP
      II=I-NP+J
```

```
   13 V(J)=DR(II)
      CALL LARAN(Z,V,N,SA,DRA)
      DO 14 J=1,NP
      II=I-NP+J
   14 V(J)=RC(II)
      CALL LARAN(Z,V,N,SA,RCA)
C  MODIF  9 2 79
      NP=2
      N=NP
C
      DO 11 J=1,NP
      II=I-NP+J
C  MODIF 8 2 79
      Z(J)=S(II)
C
   11 V(J)=DO(II)
      CALL LARAN(Z,V,N,SA,DOA)
12    CONTINUE
      RETURN
      END
C** TITRE RKGS
      SUBROUTINE RKGR(PRMT,Y,DERY,NDIM,IHLF,AUX,CLL)
      DIMENSION Y(2),DERY(2),AUX(8,2),A(4),B(4),C(4),PRMT(5),CLL(2)
      DO 1 I=1,NDIM
    1 AUX(8,I)=.06666667*DERY(I)
      X=PRMT(1)
      XEND=PRMT(2)
      H=PRMT(3)
      PRMT(5)=0.
      CALL FSY(X,Y,DERY)
C
C  ERROR TEST
C
      IF(H*(XEND-X))38,37,2
C
C  PREPARATIONS FOR RUNGE-KUTTA METHOD
C
    2 A(1)=.5
      A(2)=.2928932
      A(3)=1.707107
      A(4)=.1666667
      B(1)=2.
      B(2)=1.
      B(3)=1.
      B(4)=2.




      C(1)=.5
      C(2)=.2928932
      C(3)=1.707107
      C(4)=.5
C
C  PREPARATIONS FOR FIRST RUNGE-KUTTA STEP
C
      DO 3 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
      AUX(3,I)=0.
    3 AUX(6,I)=0.
      IREC=0
      H=H+H
      IHLF=1
      ISTEP=0
      IEND=0
C
C  START OF A RUNGE-KUTTA STEP
C
    4 IF((X+H-XEND)*H)7,6,5
    5 H=XEND-X
    6 IEND=1
C
C RECORDING OF INITIAL VALUES OF THIS STEP
C
    7 CALLDRUCK(X,Y,DERY,IREC,NDIM,PRMT,CLL)
      IF(PRMT(5))40,8,40
    8 ITEST=0
    9 ISTEP=ISTEP+1
C
C  START OF INNERMOST RUNGE-KUTTA LOOP
C
      J=1
   10 AJ=A(J)
      BJ=B(J)
      CJ=C(J)
      DO 11 I=1,NDIM
      R1=H*DERY(I)
      R2=AJ*(R1-BJ*AUX(6,I))
      Y(I)=Y(I)+R2
      R2=R2+R2+R2
   11 AUX(6,I)=AUX(6,I)+R2-CJ*R1
      IF(J-4)12,15,15
   12 J=J+1
      IF(J-3)13,14,13
   13 X=X+.5*H
   14 CALL FSY(X,Y,DERY)
      GO TO 10
C
C  END OF INNERMOST RUNGE-KUTA LOOP
C  TEST OF ACCURARY
C
   15 IF(ITEST)16,16,20
C
C  INCASE ITEST=0,THERE IS NO POSSIBILITY FOR TESTING OF ACCURACY
C
   16 DO 17 I=1,NDIM
   17 AUX(4,I)=Y(I)
      ITEST =1
      ISTEP=ISTEP+ISTEP-2
   18 IHLF=IHLF+1
      X=X-H
      H=.5*H
      DO 19 I=1,NDIM
      Y(I)=AUX(1,I)
      DERY(I)=AUX(2,I)
   19 AUX(6,I)=AUX(3,I)
      GO TO 9
C
C  INCASE ITEST=1,TESTING OF ACCURACY IS POSSIBLE
C
   20 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)21,23,21
   21 CALL FSY(X,Y,DERY)
      DO 22 I=1,NDIM
      AUX(5,I)=Y(I)
   22 AUX(7,I)=DERY(I)
      GO TO 9
C
C COMPUTATION OF TEST VALUE DELT
```

```
C
   23 DELT=0.
      DO 21 I=1,NDIM
   21 DELT=DELT+AUX(8,I)*ABS(AUX(4,I)-Y(I))
      IF(DELT-PRMT(4))28,28,25
C
C    ERROR IS TOO GREAT
C
   25 IF(IHLF-10)26,36,36
   26 DO 27 I=1,NDIM
   27 AUX(4,I)=AUX(5,I)
      ISTEP=ISTEP+ISTEP-4
      X=X-H
      IEND=0
      GO TO 18
C
C    RESULT VALUES ARE GOOD
C
   28 CALL FSY(X,Y,DERY)
      DO 29 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
      AUX(3,I)=AUX(6,I)
      Y(I)=AUX(5,I)
   29 DERY(I)=AUX(7,I)
      CALLOUTCK(X-H,Y,DERY,IHLF,NDIM,PRMT,CLL)
      IF(PRMT(5))40,30,40
   30 DO 31 I=1,NDIM
      Y(I)=AUX(1,I)
   31 DERY(I)=AUX(2,I)
      IREC=IHLF
C    MODIF  4 2 78
C     IF(IEND)32,32,39
      IF(IEND)32,32,41
   41 IHLF=11
      GO TO 39
C
C
C    INCREMENTS GET DOUBLED
C
   32 IHLF=IHLF-1
      ISTEP=ISTEP/2
      H=H+H
      IF(IHLF)4,33,33
   33 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)4,34,4
   34 IF(DELT-.02*PRMT(4))35,35,4
   35 IHLF=IHLF-1
      ISTEP =ISTEP/2
      H=H+H
      GO TO 4
C
C    RETURNS TO CALLING PROGRAM
C
   36 IHLF=11
      CALL FSY(X,Y,DERY)
      GO TO 39
   37 IHLF=12
      GO TO 39
   38 IHLF=13
   39 CALLOUTCK(X,Y,DERY,IHLF,NDIM,PRMT,CLL)
   40 RETURN
      END
C** TITRE FSY
      SUBROUTINE FSY(SA,Y,DERY)
C
C CALCULATE THE DERIVATIVES FROM S AND Y(I),I=1,2
C FOR THE COMPRESSIBLE TURBULENT BOUNDARY LAYER
C SA IS THE ARC LENGTH S/SREF
C Y(1) IS L. DERY(1)=DL/DS/SREF
C Y(2) IS X. DERY(2)=DX/DS/SREF
C DATE 2. 3. 72
C
      COMMON/A/S(32),Q(32),DQ(32),DP(32),REREF,RMREF,TREF
      COMMON/C/CF,RC(32),RCF,CLL(2),NR1,LN1
      COMMON/GAZ/N,GA,R
      COMMON/R/RE1
      DIMENSION RL(3),RMA(3)
      DIMENSION Y(2),DERY(2)
      CALL QQDSD(SA,QA,DQA,DPA,RCA)
C
C SUBROUTINE USED TO CONVERT Q INTO MACH NO. RM AND LAVAL NO. RMS
C IF THE REFERENCE LAVAL NO. RMS1 IS GIVEN
C DATE 22. 1. 72
C
      CALL QMF(GA,QA,RMREF,RMS,RM,UA)
      TA=1./F1(GA,RM)
      IF(RE1.LT.500.) RE1=500.
      H=2./3.*(1.06+ALOG(RE1))
      H=1.244/H
      H=-(H-1)
      H121=1./H
      H321=FH32(H121)
      H131=H121/H321
      RLMAX=FL(H131,RM)
      IF(Y(1)-RLMAX)1,1,2
    2 WRITE(2,3)
C*****************************************
      WRITE(6,3)
C*******************************************
    3 FORMAT(1H0,10X,'L IS GREATER THAN LMAX CALCULATION IS STOPPED')
C*******************************
C        STOP
      RETURN
C*******************************
    1 CONTINUE
      YA=Y(1)
      YB=Y(2)
      CALL SEH13(RM,Y(1),H13I,H12I,H13)
      RMA(2)=RM
      RMA(1)=RM-.05
      RMA(3)=RM+.05
      RL(2)=Y(1)
      RL(1)=FL(H13I,RMA(1))
      RL(3)=FL(H13I,RMA(3))
      DLDMA=(RL(2)-RL(1)+RL(3)-RL(2))/(.05*2.)
      DMDSA=.5*DUA/TA
      AFA=(R-N)*(GA-1.)*RM**2/2.
      DADMA=(R-N)*(GA-1.)*RM
      DADSA=DADMA*DMDSA
      TA=(TREF/TA)**(1.+N-GA/(GA-1.))
      CALL SBTCT(RM,YA,YB,BT,CT,RCA,GA,DUA)
```

```
      BIMAX=2.
      IF(CT)1,6,6
    1 IF(BI)6,7,7
    7 BTI=-ABS(BI)
    6 BTI=BI
      IF(ABS(BTI)-BIMAX)8,9,9
    9 BTI=SIGN(BIMAX,BTI)
    8 DERY(2)=GA*REREF*TA/EXP(YE)*CT-DRA
      DERY(1)=DGA-BTI*(DERY(2)+DRA)
      DERY(1)=DERY(1)+DLCMA*DMDSA
      DERY(2)=DERY(2)+2.*(1.+AFA)*DLCMA*DMDSA+2.*YA*DADSA
      RETURN
      END
C** TITLE HUCLF
      SUBROUTINE HUCLF
      COMMON/A/S(32),Q(32),DG(32),DR(32),REREF,RMS,TREF
      COMMON/B/PRMT(5),Y(2),DERY(2),NDIM,IHLF
      COMMON/C/CF,FC(32),NCF,DUM(2),NF1,LNF
      COMMON/GAZ/W,GA,R
      COMMON/ESPC/ DUMP(487), IPRNT, DUM1(4), CDV(480)
      DIMENSION CLL(2)
      DIMENSION AUX(8,2)
      NCF=1
      N=LNF
      DO 32 I=1,N
      IF(Q(I))10,10,12
   10 Q(I)=0.01
   12 CONTINUE
   32 Q(I)=ALOG(Q(I))
      CALL DERIR(N,S,Q,DQ)
      IF(IPRNT.EQ.1) WRITE(2,33)REREF,RMS
   33 FORMAT(/1X,'REREF=',F10.0,5X,'RMCREF=',F10.5)
      IF(IPRNT.EQ.1) WRITE(2,601)
  601 FORMAT(//6X,'S/SREF',1X,'U/UREF',5X,'DU',9X,'L',9X,'X',7X,'H12I',
     17X,'H12',9X,'RE2',10X,'DEL1',10X,'DEL2',10X,'DEL3'/)
C
C   RM IS THE REFERENCE MACH NO
C   TREF IS TREF/T-TOTAL
      RM=SQRT(2.*RMS**2/((GA+1.)-(GA-1.)*RMS**2))
      TREF=1./(1.+(GA-1.)*RM**2/2.)
      IHLF=1
      CLL(1)=DUM(1)
      CLL(2)=DUM(2)
      CALL RKGR(PRMT,Y,DERY,NDIM,IHLF,  AUX,CLL)
      DUM(1)=CLL(1)
      DUM(2)=CLL(2)
      IF(IHLF-11)22,23,24
   23 WRITE(2,201)
      GO TO 21
C   MODIF  8 2 74
   24 IF(IHLF-13)25,26,27
C   24 IF(IHLF-13)25,26,26
C
   25 WRITE(2,301)
      GO TO 21
   26 WRITE(2,401)
      GO TO 21
C   MODIF  8 2 78
   27 IF(IPRNT.EQ.1) WRITE(2,701)
  701 FORMAT(/,1X,'END OF CALCULATION',/)
C
   22 IF(IPRNT.EQ.1) WRITE(2,501)PRMT(2),Y(1),Y(2)
   21 CONTINUE
  201 FORMAT(/' MORE THAN 10 BISECTIONS'/)
  301 FORMAT(/' INCREMENT OR INTERVAL EQUAL TO ZERO'/)
  401 FORMAT(/' INCREMENT HAS THE WRONG SIGN'/)
  501 FORMAT(/' S=',F9.5,2X,2HL=,E12.5,2X,2HX=,E12.5/)
      RETURN
      END
C  ERG
C INTERPOLATION FUNCTION SUBROUTINE FOR CT AND MT
C XMAX=14.
C MAXIMUM MACH NO. = 1.5
C
      FUNCTION ERG(NA,RRM,Z,XX)
C
C NA=1, CALCULATION OF MT
C NA=2, CALCULATION OF CT
C
C RRM= MACH NO.
C Z=L OR SQRT(L)
C XX=X
C
C
C
C     SUBROUTINES ERG  RESUL  LARAN
C
C
      COMMON/GAZ/W,GA,R
      DIMENSION FM(4),Y(4),FX(6),YZ(2)
      FM(1)=.001
      FM(2)=.5
      FM(3)=1.
      FM(4)=1.5
      NM=4
      FX(1)=4.
      FX(2)=6.
      FX(3)=8.
      FX(4)=10.
      FX(5)=12.
      FX(6)=14.
      NXMAX=6
C
C IF XX=FX(NZ), THEN NN=1, NO INTERPOLATION NECESSARY, OTHERWISE NN=2,
C USE LINEAR INTERPOLATION
C
      NN=2
C
C TEST IF RRM LESS THAN .001. IF YES, CALCULATE AS IF IT WERE .001
C
      IF (RRM-0.001) 7,7,8
    7 RM=.001
      GO TO 9
    8 RM=RRM
C
C TEST IF X LESS THAN 4. IF YES, CAL CULATE AS IF IT WERE 4
C
    9 IF(XX-4.)5,5,6
    5 NZ=1.
      NN=1
      IX=NZ
      X=4.
      GO TO 40
```

```
    6 X=AX
      NZ=1
  101 NZ=NZ+
      IF (NZ-NXMAX) 104,104,105
  105 WRITE(2,106)
C*****************************************
      WRITE(6,106)
C*****************************************
  106 FORMAT(1X,43HX-VALUE GREATER THAN 14, CALCULATION STOPPED,/)
C*****************************
C       STOP
      RETURN
C*****************************
  104 IF (X-FX(NZ)) 102,103,101
  103 NN=1
      GO TO 100
  102 NZ1=NZ-1
  100 CONTINUE
      K=0
      GO TO (31,32),NN
   31 IX=NZ
      GO TO 40
   32 DO 3 IX=NZ1,NZ
      K=K+1
   40 DO 2 IM=1,NM
      GO TO (21,22),NA
   21 Y(IM)=RESUL(1,IX,IM,Z)
      A=(R-1)*(GA-1.)*FM(IM)**2
      A=-0.5/(1.+A)
      IF (Y(IM)-A) 61,2,2
   61 ZZ=Z
   64 ZZ=ZZ-.01
      B=RESUL(1,IX,IM,ZZ)
      IF (B-A) 64,65,65
   65 Z1=ZZ-.001
      A=RESUL(1,IX,IM,Z1)
      Y(IM)=(A-B)/(Z1-ZZ)*(Z-ZZ)+B
      GO TO 2
   22 Y(IM)=RESUL(2,IX,IM,Z)
    2 CONTINUE
C   MODIF 30 6 2 78
C     CALL LAGRAO(FM,Y,NM,RM,YN)
      CALL LAGRAN (FM,Y,NM,RM,YN)
C
      GO TO (51,52),NN
   51 ERG=YN
      GO TO 60
   52 YZ(K)=YN
    3 CONTINUE
C
C LINEAR INTERPOLATION BETWEEN FX(NZ1) AND FX(NZ)
C
      ERG=YZ(1)+(YZ(2)-YZ(1))/(FX(NZ)-FX(NZ1))*(X-FX(NZ1))
   60 RETU
      END
C   SFH13
      SUBROUTINE SFH13(RM,RL,H131,H121,H13)
      COMMON/ESFC/ DUMMU(487), IPRNT, DUMT(4), CDV(480)
C
C     ITERATION TO CALCULATE H13I
C
C    MODIF  26 4 78
      IT=1
C
      RMB=RM*RM
      RMC=RMB*RM
      RMD=RMC*RM
      RME=RMD*RM
      RMF=RME*RM
      A=0.7973151+0.6650299E-3*RM-0.6994673*RMB-0.244131E-1*RMC
     1 +0.4165987*RMD-0.2126298*RME+0.3347239E-1*RMF
      B=-0.4130449-0.7119359E-3*RM+0.3720671*RMB+0.1080879E-1*RMC
     1 -0.2229626*RMD+0.1139661*RME-0.1795971E-1*RMF
      C=0.4291051E-1+0.3360753E-3*RM-0.6995547E-1*RMB-0.1737653E-2*RMC
     1 +0.4300749E-1*RMD-0.2215015E-1*RME+0.3501069E-2*RMF
      D=-0.4475711-0.7629619E-3*RM+0.5607572*RMB+0.2373655E-1*RMC
     1 -0.3396425*RMD+0.171969*RME-0.2693042E-1*RMF
      E=0.3362769-0.6434467E-3*RM-0.187885*RMB-0.7611719E-2*RMC
     1 +0.1032432*RMD-0.5064087E-1*RME+0.7761412E-2*RMF
      IF(RL-.0062) 21,22,22
   21 H13I=1.9343
      GO TO 24
   22 CONTINUE
C     H=1.35     OLD
      H  =2.05-(10.*RL)**.33
   23 F=A+B*H+C*H*H+D/H+E/(H*H)-SQRT(RL)
      DF=B+2.*C*H-D/(H*H)-2.*E/(H**3)
      CH=H
C     H=H-F/DF
C    MODIF 26 4 78
C     H=H-.5*F/DF         NEW 76
      H=H-.25*F/DF
      IT=IT+1
      IF(IT-300)26,26,27
   27 IF(IPRNT.EQ.1) WRITE(2,28)H,RL
   28 FORMAT(2E14.5)
       H131=.5
      GO TO 24
   26 CONTINUE
      H=ABS(H)
C
      IF(ABS(F*SQRT(RL)).GT.5.E-3) GO TO 23
      H13I=H
   24 CONTINUE
C
C     ITERATION TO CALCULATE H12I
      X=1.977
   25 X2=X*X
      X3=X2*X
      X4=X3*X
      X5=X4*X
      P=0.4422849E01-0.4698586E01*X+0.3287947E01*X2-0.1227710E01*X3
     1+0.2380595*X4  -0.1878198E-1*X5-X/H13I
      DP=-4.698586-1./H13I+6.575894*X-3.68313*X2+0.952238*X3-.0939099*X4
      CX=X
      X=X-P/DP
      IF(ABS(X-CX).GT.1.E-3) GO TO 25
      H12=FN(X,RM)
      H32I=X/H13I
      H32=H32I*(1.+.011*RM/1.5)
      H13=H12/H32
      H12I=X
```

Copy available to DDC does not permit fully legible reproduction

```
      RETURN
      END
      SUBROUTINE HRA(H121,DUM,RLA)
      COMMON/R/RE1
      A=(1.-1./H121)/).41
      B=1.55+ALOG(RE1)
      C=0.3714-0.205*A
      D=1.15-0.2733*A-0.205*B*A
      E=0.205*A
      F=1.2444-0.2733*A*B
      G=0.2733*A
      RLA=12.0
      N=1
    4 H=1.2195*RLA+1.626
      P=C*RLA**2+D*RLA+E*RLA*ALOG(H)+G*ALOG(H)+F
      Q=2.*C*RLA+1.2195*(E*RLA+G)/H+E*ALOG(H)+D
      DLAN=P/Q
      IF(DLAN.LT.3.0) GO TO 9
      DLAN=3.0
    9 CONTINUE
      RLAN=RLA-DLAN*.5
      IF(RLAN)10,11,11
   10 RLAN=0.
   11 CONTINUE
      IF(ABS(RLA-RLAN).LT.1.E-3.OR.N.GT.100) GO TO 7
      N=N+1
      RLA=RLAN
C     GO TO 4    OLD
      IF(N.LT.90)GO TO 4
    7 CONTINUE
      RETURN
      END
C** FIRE HRLST
C   MODIF DU 8 2 78
C     SUBROUTINE HRLST(H32,DSS,U,TURBP,IDIR,X)
      SUBROUTINE HRLST(H32,DSC,U,TURBP,IDIR,X)
C
      COMMON/A/XD(32),UD(32),DUM(64),RE,AM1,DYM
      COMMON/B/XSTAG,DIM(5),KDIM(2)
      COMMON/C/DUMY(33),JDIM,DSTC,THCC,NR1,LNT
      COMMON/GAZ/+,G,KGIR
      COMMON/FSFC/ DUINT(487), IPRNT, DUMT(4), CDV(480)
      COMMON/GSFC/ DELTS(480), SDELTS(480), CODEL(480), DCUDS(480), IDS,
     1IBETA(480)
      COMMON/SAVE/ TEST1, TEST2, IDUMI
      DATA IBLAN,ISTAR/' ',' *'/
C*****************************************
C      DATA TEST1,TEST2/0.0001,0.01/
C*****************************************
      NT=LNT
      NINT=3
      IN=0
      ITEST=IBLAN
      ISTOP=2
      IN=IN+1
      GU=(G-1.)/(G+1.)
      GP=(G-1.)/2.
      GUP=1./(GU*2.)
      GUS=G/(G-1.)
      AMC12=AM1*AM1*(G+1.)/(2.+(G-1.)*AM1*AM1)
      TEST1=TEST1*XD(NT)+1.E-7
      TEST2=TEST2*XD(NT)+1.E-7
      IMPZ=2
      IF(TURBP)72,72,71
   71 SS=1.126/EXP(1.2*TURBP)
      GO TO 73
   72 SS=1.E5
   73 I=0
   74 I=I+1
      NJ=I +1
      X=XD(I)
      U=UD(I)
      IF((X-XSTAG)*ABS(U)-0.005)74,75,75
   75 CALL HRIND (N1,XD,X,UD,U,DU,NINT)
      DUX=DU/U
      X1=X-XSTAG
      ATRA=0.
      DALS=0.
      R=1.
      AM=DUX*X1
      IF(AM-1.)83,83,80
   80 AM=1.
   83 IF(AM-0.1)81,82,82
   81 AM=0.1
   82 AL=0.0467-0.0032/AM+1.2E-4/AM**2
      CALL COLEL(AL,R,RE,H13,H12,B,C)
      R=SQRT(2.*C*U*X1*(1.-GU*AMC12)**.5*(1.-GU*U*U*AMC12)**GUS/(AM+1.))
      CALL COLEL(AL,R,RE,H13,H12,B,C)
      AL3=AL
      DSS=R/(U*((1.-GU*AMC12)/(1.-GU*AMC12*U*U))**.5)/EXP(2.*AL)
      DST=H13*DSS
      THC=DST/H12
      XX=ALOG(R*RE)
      AML2=AMC12*U*U/(1.+GP*(1.-AMC12*U*U))
      FMU=1.+GP*AML2
      DSTC=(FMU**GUP)*(FMU*DST+GP*AML2*THC)
      THCC=FMU**GUP*THC
      H12C=FMU*H12+GP*AML2
      IF(IPRNT.EQ.1) WRITE(2,106)      RE,TURBP,AM1
      IF(IPRNT.EQ.1) WRITE(2,101)ITEST,X,U,DU,DSTC,THCC,H12C,AL,XX
      DX=0.02*X1*U/AM
   91 X1=X
      IDIR=1
      DUX1=DUX
      U1=U
      T1=THC
      ALF=AL
      XX1=XX
      BB=B
      CC=C
   65 X=X1+DX
      C=CC
      B=BB
      DX1=DX
      XI=X1+DX1/2.
      CALL HRIND (NT,XD,XI,UD,UI,DUI,NINT)
      DUXI=DUI/UI
      CALL HRIND (NT,XD,X,UD,U,DU,NINT)
      DUX=DU/U
      DR1=C*U1*DX1*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*U1*U1)**GUS
      DL1=DUX1*DX1-B*DR1
      R1=R+DR1/2.
```

```
      AL1=AL+DL1/2.
      CALL COLEL(AL1,R1,RE,H13,H12,B,C)
      DR2=C*UI*DX1*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*UI*UI)**GUS
      DL2=DUX1*DX1-B*DR2
      R2=R+DR2/2.
      AL2=AL+DL2/2.
      CALL COLEL(AL2,R2,RE,H13,H12,B,C)
      DR3=C*UI*DX1*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*UI*UI)**GUS
      DL3=DUX1*DX1-B*DR3
      R3=R+DR3
      AL3=AL+DL3
      CALL COLEL(AL3,R3,RE,H13,H12,B,C)
      DR4=C*U*DX1*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*U*U)**GUS
      DL4=DUX*DX1-B*DR4
      AL1=AL+(DL1+2.*DL2+2.*DL3+DL4)/6.
      RTE=R+(DR1+2.*DR2+2.*DR3+DR4)/6.
      IF(ABS(ALI-AL3)*DX/XD(NT)-1.E-7)23,24,24
   23 DX=DX*1.5
      DXMAX=RE*R*R/10.
      IF(DX-DXMAX)662,662,661
  661 DX=DXMAX
  662 IF(DX-TESTB)66,66,64
   64 DX=TESTB
      GO TO 66
   24 DX=DX*0.5
      IF(DX-TESTA)61,65,65
   61 DX=TESTA
   66 AL=ALI
      R=RTE
      CALL COLEL (AL,R,RE,H13,H12,B,C)
      HIS=B*SQRT((1.-GU*AMC12)/(1.-GU*AMC12*U*U))
      DSS=R/HIS/EXP(3.*AL)
      DST=H13*DSS
      THC=DST/H12
      H32=DSS/THC
      RT=RE*UIU*THC
      DTX=(THC-T1)/DX1
      XX=ALOG(R*RE)
      IF(X-XL(NW))10,20,20
   20 NTL=1
   10 AML2=AMC12*U*U/(1.+GP*(1.-AMC12*U*U))
      FMU=1.+GP*AML2
      DSTC=(FMU**GUP)*(FMU*DST+GP*AML2*THC)
      THCC=FMU**GUP*THC
      H12C=FMU*H12+GP*AML2
      IF(IPRNT.EQ.1) WRITE(2,101)ITEST,X,U,DU,DSTC,THCC,H12C,AL,XX
C*************************************
      IF(X.EQ.SDELTS(IDS)) GO TO 223
      IDS=IDS+1
      DELTS(IDS)=DSTC
      CUDEL(IDS)=U
      SDELTS(IDS)=X
      DCUDS(IDS)=DU
      THETA(IDS)=THCC
      DELVEL=CUDEL(IDS)-CUDEL(IDS-1)
      DELTHE=(THETA(IDS)-THETA(IDS-1))/(CUDEL(IDS)-CUDEL(IDS-1))
      COV(IDS)=2.0*(DELTHE+THCC/U*(2.0+H12-AML2**0.5)**2*DU)*U**2
  223 CONTINUE
C*************************************
      GO TO (131,142,164),IDIR
  131 IF(X-XL(NT))161,161,999
  181 IF(AL)15,15,13
   15 IF(DU-0.2)182,182,671
   13 DALS1=DALS
      DALS=0.005*XX-0.011-AL
      TPR=3.1E-7/(R*RE)
      ITEST=IBLAN
      IF(DALS)151,151,14
  151 ATRA=0.
      ISCHT=1
      GO TO 91
   14 ITEST=ISTAR
      GO TO (141,142),ISCHT
  141 ISCHT=2
      IDIR=2
      RTI=RT
      AK=RE*THC*THC*DU*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*U*U)**((2.*G-1.)/(
     1G-1.))
      XI1=X-1.E-5
      XIL=1.E-5
      X1=X
  142 AK1=AK
      AK=RE*THC*THC*DU*SQRT(1.-GU*AMC12)*(1.-GU*AMC12*U*U)**((2.*G-1.)/(
     1G-1.))
      ATRA=ATRA+0.5*(AK+AK1)*(X-X1)
      XIL=XIL+(X-X1)*(1.-GU*AMC12*U*U)**((3.*G-1.)*.5/(G-1.))
      X1=X
      AKM=ATRA/XIL
      IF(AKM)1421,1422,1422
 1421 RTT=RTI+SS*400.*(EXP(80.*AKM)+1.)
      GO TO 1423
 1422 RTT=RTI+SS*32./(0.04-AKM)
 1423 IF(RT-RTT)91,163,163
  163 IDIR=3
      GO TO 10
  182 AL=0.
      WRITE(2,103)
C********************************
      WRITE(6,103)
C********************************
      H32=1.51495
  164 IF(IPRNT.EQ.1) WRITE(2,104)
C  MODIF LU 6 2 78
      DSC=DSS*FMU**GUP
      IF(IPRNT.EQ.1) WRITE(2,102)X,H32,DSC,U
C     WRITE(2,102)X,H32,DSS,U
C
      IDIR=2
      GO TO 1000
  671 IDIR=3
      WRITE(2,109)
C***************************
      WRITE(6,109)
C***************************
      GO TO 1000
  999 IDIR=1
      IF(IPRNT.EQ.1) WRITE(2,108)
 1000 RETURN
  106 FORMAT(///' LAMINAR BOUNDARY LAYER'/' LEFOLL + STEWARTSON'/
     1' RE=',F9.0,'  TURB = ',F5.2,' PERC.','INLET MACH NUMBER = ',F7.3/
     26X,1HS,11X,1HU,10X,5HDU/DS,8X,2HD1,10X,2HD2,10X,3HH12,9X,1HL,11X,1
     3HX/)

  101 FORMAT(A2,3(F9.4,3X),2E12.3,F9.4,3X,E12.3,F9.4)
  109 FORMAT(//' THE PROGRAM DIVERGES AND IS STOPPED')
  102 FORMAT(//' CONDITIONS AT TRANSITION POINT'//,F10.5,2E12.3,F10.5)
  103 FORMAT(/' LAMINAR SEPARATION'/)
  104 FORMAT(/' TRANSITION'/)
  108 FORMAT(/' FULL LAMINAR  B.L.'/)
```

Copy available to DTIC does not permit fully legible reproduction

```
      END
C   LARAN  (VERSION CULCX)
      SUBROUTINE LARAN(X,Y,NI,XI,YI)
      DIMENSION X(4),Y(4)
      J=0
      NP=NI
    1 NP=NP-1
      J=J+1
      DO 2 K=1,NP
      I=J+K
    2 Y(K)=(Y(K)*(X(I)-XI)+Y(K+1)*(XI-X(K)))/(X(I)-X(K))
      IF (NP-1) 3,3,1
    3 YI=Y(1)
      RETURN
      END
C** TITRE DERIR
      SUBROUTINE DERIR(N,S,W,DER)
      DIMENSION Y(5),X(5),XX(5),YY(5),A1(5),A2(3),A3(2),S(1),W(1),DER(1)
      M=0
      K=1
      M1=0
      II=0
      JJ=1
   17 DO 10 I=JJ,5
      L=I+K-1
      XX(I)=S(L)
   10 YY(I)=W(L)
   16 M=M+1
      M1=M-L+5
      X(1)=XX(M1)
      Y(1)=YY(M1)
      M3=M1-1
      IF(M3)28,28,18
   18 CONTINUE
      DO 1 I=1,M3
      K1=M1-I
      X(M1+1)=XX(K1)
      Y(M1+1)=YY(K1)
    1 CONTINUE
   28 CONTINUE
      M2=M1+1
      IF(M2-5)20,21,21
   20 DO 19 I=M2,5
      X(I)=XX(I)
   19 Y(I)=YY(I)
   21 DO 2 I=1,4
    2 A1(I)=(Y(I)-Y(I+1))/(X(I)-X(I+1))
      DO 3 I=1,3
    3 A2(I)=(A1(I)-A1(I+1))/(X(I)-X(I+2))
      DO 4 I=1,2
    4 A3(I)=(A2(I)-A2(I+1))/(X(I)-X(I+3))
      A4=(A3(1)-A3(2))/(X(1)-X(5))
      II=II+1
      DER(II)=A1(1)-(X(2)-X(1))*A2(1)+(X(2)-X(1))*(X(3)-X(1))*A3(1)-(X(2
     1)-X(1))*(X(3)-X(1))*(X(4)-X(1))*A4
      JJ=3
      IF(M-2)22,22,23
   23 IF(M-N+2)24,22,22
   22 IF(M-N)26,27,27
   26 GO TO 16
   24 DO 25 I=1,2
      XX(I)=XX(I+1)
   25 YY(I)=YY(I+1)
      K=K+1
      GO TO 17
   27 RETURN
      END
C** TITRE FFCT
      FUNCTION FFCT(RM,RL,RX)
C
C FUNCTION USED TO CALCULATE CT
C DATE 1/7/71
C
      IF(RL)1,1,2
    1 WRITE(2,4)RL
    4 FORMAT(1X,F15.12)
      RL=.0002
    2 CONTINUE
      Z=SQRT(RL)
      FFCT=ERG(2,RM,Z,RX)
      RETURN
      END
C** TITRE HRIND
      SUBROUTINE HRIND (N,X,XI,Y,YI,CYI,NI)
      DIMENSION X(1),Y(1),XF(10),YF(10)
      DIMENSION D1(10)
      IF(NI-N) 3,3,2
    2 NI=N
    3 IF(XI-X(1))7,8,8
    7 IRES=1
      GO TO 5
    8 DO 1 I=2,N
      IF(XI-X(I-1))1,4,4
    4 IF(XI-X(I))14,1,1
   14 IRES=I
      GO TO 5
    1 CONTINUE
      IF(XI-X(N))5,9,9
    9 IRES=N
    5 NR=0
      NL=0
      NS=NI/2+1
      NP=NI-NS
      IF(IRES-NS)6,10,10
    6 NR=NS-IRES
      GO TO 12
   10 IF(IRES-N+NP)12,12,11
   11 NL=IRES+NP-N
   12 DO 13 I=1,NI
      L=I+IRES-NS+NR-NL
      XF(I)=X(L)
   13 YF(I)=Y(L)
      J=0
      NP=NI
   15 NP=NP-1
      J=J+1
      DO 16 K=1,NP
      I=K+J
   16 YF(K)=(YF(K)*(XF(I)-XI)+YF(K+1)*(XI-XF(K)))/(XF(I)-XF(K))
      IF(NP-1)17,17,15
   17 YI=YF(1)
      ID=0
```

```
      NF=NI-1
      DI(1)=YI
      DO 30 I=1,NI
      L=I+INES-NS+NR-NL
      II=I-IE
      IF(ABS(X(L)-XI)-1.E-4)41,41,42
41    IE=1
      NE=NE-1
      GO TO 30
42    XF(II)=X(L)
      YF(II)=Y(L)
30    CONTINUE
      DO 31 K=1,NE
      IS=NF-K+1
      DI(K+1)=(DI(K)-YF(1))/(XI-XF(K))
      DO 31 I=1,IS
      IK=I+K
31    YF(I)=(YF(I)-YF(I+1))/(XF(I)-XF(IK))
      DYI=(XI-XF(NE))*(DI(NE+1) -YF(1))/(XI-XF(NE+1))
      K=1
33    K=K+1
      IF(K-NE)34,34,32
34    IS=NE-K+2
      DYI=(DYI+DI(IS+1))*(XI-XF(IS-1))
      GO TO 33
32    DYI=DYI+DI(2)
      RETURN
      END
      SUBROUTINE CCLEL(AL,R,RE,H13,H12,B,C)
      COMMON/SAVE/ TEST1, TEST2, IDUMI
      DIMENSION ALA(7), BLA(7), CLA(7),ATU(5,7),BTU(5,7),AT(7),BT(7)
      DIMENSION PBLA(7)
      DATA ALA/+7.850623E-1,-8.704517E00,-1.623161E2,4.650856E3,-7.60999
     C E4,3.545592E5,0./,BLA/3.285779E-1,-1.564922E1,-3.264378E2,1.87327
     C E3,-3.902193E5,3.845798E6,-1.491445E7/,CLA/2.662485E0,-1.129277E1
     C ,1.801993E1,-2.062866E2,7.341341E2,-1.169111E3,0./
      DATA PBLA/ 0.177694,-11.5538,5*0./
C*******************************************
C******************************************
      IF(AL.LT.0.)AL=0.
      SAL=SQRT(AL)
1     IF(AL-0.75)82,82,81
81    DO 93 K=1,7
83    BLA(K)=PBLA(K)
92    C=ALA(1)
      B=BLA(1)
      H13=CLA(1)
      SALE=1.
      DO 11 J=2,7
      SALE=SALE*SAL
      I=J-1
      C=C+ALA(J)*SALE*SALE
      B=B+BLA(J)*SALE*SALE
11    H13=H13+CLA(J)*SALE
      B=-B/(1.+2.*B)
      B=B/R
      C=C/(R*RE)
      IF(AL-1.E-3)441,441,442
441   H12=1.03
      H13=2.66
      RETURN
442   H12=1.42/AL**0.15
      RETURN
      END
C   OMF
      SUBROUTINE OMF(GA,O,RMS1,RMS,RM,U)
C
C SUBROUTINE USED TO CONVERT O INTO MACH NO. RM AND LAVAL NO. RMS
C IF THE REFERENCE LAVAL NO. RMS1 IS GIVEN
C DATE 22. 1. 72
C
      U=EXP(O)
      RMS=U*RMS1
      RM=FFM(GA,RMS)
      RETURN
      END
      SUBROUTINE SF(RMM,RLA,FU,RE,CA,CB,RNU,DEL,F)
      COMMON/ESFC/ DUMME(487), IPRNT, DUMI(4), CDV(480)
C
C SUBROUTINE FOR CALCULATION OF THE SKIN FRICTION FOR THE COMPRESSIBLE
C FLOW UP TO M=1.5
C VERSION 5. 3. 73.
C MODIFICATION TO ACCELERATE CONVERGENCE 31. 7. 75.
C
      CALL SDEL(RLA,FU,RMM,DEL)
      CR=(ALOG(RE*RNU/DEL)+RLA-0.5)/0.41+5.
      AN=0.
      F=1.
5     CL=CB/(CA*F)-ALOG(F)/.41
      DC=CL-CR
      D=ABS(DC)-.00001
      AN=AN+1.
      ITER=IFIX(AN)
      IF(ITER-22)6,7,7
7     IF(IPRNT.EQ.1) WRITE(2,8)ITER,CA,CB,F,CL,CR,DC
8     FORMAT(1X,'ITER=',I3,6F12.6)
      GO TO 1
6     CONTINUE
      DD=D/CR
      IF(DD-.001)1,1,2
2     DF=(CR-CL)*F*F/(CB/CA+F/.41)
      DF=DF*((AN/(AN+1.))**2.)
      F=F-DF
      IF(F)9,9,5
9     F=.0001
      GO TO 5
1     CONTINUE
      RETURN
      END
      SUBROUTINE SDEL(RLA,FU,RMM,DEL)
C
C SUBROUTINE TO CALCULATE THE DISPLACEMENT THICKNESS FOR COMPRESSIBLE
C TURBULENT BOUNDARY LAYER USING DEFOLL'S PROFILE
C
      DIMENSION U(50),ADEL(50),X(50)
      N=49
      RN=FLOAT(N)
      DO 101 I=1,N
      RI=FLOAT(I)
101   X(I)=(RI-1.)/(RN-1.)
      U(1)=0.
      DO 49 I=2,N
```

Copy available to DTIC does not permit fully legible reproduction

```
      U(I)=VPLI(FU,X(I),RLA)
      J=1./(1.+RM4*(1.-U(I)**2))
 49   U(I)=1.-U*J(I)
      CALL SMPSN(N,X,U,ADEL)
      DEL=ADEL(N)
      RETURN
      END
C SMPSN
      SUBROUTINE SMPSN(N,X,Y,S)
      DIMENSION X(50),Y(50),S(50) ,K(2)
      S(1)=0.
      S(2)=(Y(1)+Y(2))*(X(2)-X(1))/2.
      M=N-1
      K(1)=2
      K(2)=3
      DO 10 I=1,2
      L=K(I)
      DO 10 J=L,M,2
      T1=3.*X(J)-X(J+1)-2.*X(J-1)
      T2=4.*(X(J+1)-X(J-1))
      T3=3.*X(J+1)+X(J-1)-3.*X(J)
 10   S(J+1)=S(J-1)+(Y(J-1)*T1+Y(J)*T2+Y(J+1)*T3)/6.
      RETURN
      END
      FUNCTION VPLI(F,X,RLA)
      G1=(X*X -1.)/2.-ALOG(X)
      G2=2.*X*(X-3.*X*X +1.
      VPLI=1.-F*(G1+RLA*G2)/0.41
      RETURN
      END
C   FFM
      FUNCTION FFM(GA,RMS)
C
C FUNCTION USED TO CONVERSE THE LAVAL-NO. RMS TO THE MACH NO. FFM
C
      FFM=SQRT(2.*RMS*RMS/((GA+1.)-(GA-1.)*RMS*RMS))
      RETURN
      END
      SUBROUTINE SAB(RM,H,A,B)
C SUBROUTINE USED TO FIND THE COEFFICIENTS A,B FOR THE
C CURVATURE CORRECTION OF THE DISSIPATION FACTOR
C INPUT=RM(MACH NO.),H=H121
C
C OUTPUT=A,B
C VERSION 12. 6.72
C MODIFIED TO ACCELERATE
C
      H1=H
      H2=H*H1
      H3=H*H2
      H4=H*H3
      H5=H*H4
      H6=H*H5
      A=.7976361E01-.3376311E02*H1+.4591844E02*H2-.2808928E02*H3
     1+.9079914E01*H4-.1495455E01*H5+.1004765*H6
      A=A*.001
      B=-.1362206E01-.1561366E02*H1+.4774304E02*H2-.4415723E02*H3
     1+.1936466E02*H4-.4155172E01*H5+.3515623*H6
      B=(B+.467*RM*RM)*.001
      RETURN
      END
C  FT
      FUNCTION FT(GA,RM)
C
C FUNCTION USED TO CALCULATE THE RATIO OF TOTAL TEMPERATURE TO STATIC
C TEMPERATURE WHEN THE MACH NO. RM IS GIVEN
C
      FT=1.+(GA-1.)*RM*RM/2.
      RETURN
      END
C  RESUL
      FUNCTION RESUL(N,IX,IM,Z)
C
C N=1 CALCULATION FOR MT, Z=L
C N=2 CALCULATION FOR CT, Z=SQRT(L)
C THE COEFFICIENTS FOR MT AND CT ARE STORED AS FILE AND ARE READ FROM
C THE DISK WHEN USED
C DATE 10/7/71
C
      COMMON/TABLE/TAB1(336)
      DIMENSION A(7)
      IF (N-1) 2,1,2
    1 ISTART=((IX-1)*4+(IM-1))*7
      GO TO 5
    2 ISTART=(24+((IX-1)*4+(IM-1)))*7
    5 IA=ISTART
      DO 6 J=1,7
      IA=IA+1
    6 A(J)=TAB1(IA)
      RESUL=A(1)
      PROD=1.
      DO 3 J=2,7
      PROD=PROD*Z
    3 RESUL=RESUL+A(J)*PROD
      RETURN
      END
C  FFBT
      FUNCTION FFBT(RM,RL,RX)
      COMMON/GAZ/W,GA,R
C FUNCTION USED TO CALCULATE BETAT
C DATE 1/7/71
C
      RMT=ERG(1,RM,RL,RX)
C
C R=RECOVERY FACTOR
C W=EXPONENT FOR THE POWER LAW OF VISCOSITY
C GA=ISENTROPIC EXPONENT OF THE GAS
C THE FOLLOWING VALUES OF R,W,AND GA ARE FOR AIR
      A=0.5*(R-W)*(GA-1.)*RM*RM
      FFBT=-RMT/(1.+2.*(1.+A)*RMT)
      RETURN
      END
C  FH32
      FUNCTION FH32(X)
C
C CALCULATE H32I FROM H12I FOR INCOMPRESSIBLE TURBULENT BOUNDARY LAYER
C
      X2=X*X
      X3=X2*X
      X4=X3*X
      X5=X4*X
      FH32   =0.4422849E01-0.4698586E01*X+0.3287947E01*X2
```

```
      1 -0.1227710EC1*X3  +0.2380595*X4  -0.1876198E-1*X5
        RETURN
        END
C   FH
        FUNCTION FH(AHI,RM)
C
C  FH IS THE H12 FOR COMPRESSIBLE TBL (VALID FOR M=1.5 OR LESS)
C  AHI=H12I
C  RM=MACH NO.
C  VERSION 30. 6. 71
C
        FH=AHI*(1.+RM*RM*.1145)+1.55*.176*RM*RM
        RETURN
        END
C   FL
        FUNCTION FL(H,RM)
C
C  CALCULATION OF L. H=H13I, RM=MACH NO.
C
        RMB=RM*RM
        RMC=RMB*RM
        RMD=RMC*RM
        RME=RMD*RM
        RMF=RME*RM
        A=0.7773151+0.6559299E-3*RM-0.6994673*RMB-0.244131E-1*RMC
      1 +0.4185987*RMD-0.2126296*RME+0.3347239E-1*RMF
        B=-0.1130449-0.7119359E-3*RM+0.3720671*RMB+0.1080879E-1*RMC
      1 -0.2229525*RMD+0.1139061*RME-0.1799716E-1*RMF
        C=0.1291051E-1+0.3300753E-3*RM-0.6995547E-1*RMB-0.1737853E-2*RMC
      1 +0.3360709E-1*RMD-0.2215015E-1*RME+0.3504099E-2*RMF
        D=-0.4475711-0.7629619E-3*RM+0.5607572*RMB+0.237365E-1*RMC
      1 -0.3396425*RMD+0.171369*RME-0.2693042E-1*RMF
        E=0.3362765-0.6934167E-3*RM-0.167865*RMB-0.7641719E-2*RMC
      1 +0.1032132*RMD-0.5564087E-1*RME+0.7761412E-2*RMF
        A=A+B*H+C*H*H+D/H+E/(H*H)
        FL=A*H
        RETURN
        END
```

CFC1 IN

```
010507091618
NACA 2412  TEST CASE
6.2832   0.0
018064
    0.000000  0.000000  0.000000   0.00000
    0.0125    0.02150   0.0125    -0.01650
    0.0250    0.02990   0.0250    -0.02270
    0.0500    0.04130   0.0500    -0.03010
    0.0750    0.04960   0.0750    -0.03460
    0.1000    0.05630   0.1000    -0.03750
    0.1500    0.0661    0.1500    -0.04100
    0.2000    0.0726    0.2000    -0.04230
    0.2500    0.07670   0.2500    -0.0422
    0.3000    0.07880   0.3000    -0.04120
    0.4000    0.07800   0.4000    -0.03800
    0.5000    0.07240   0.5000    -0.0334
    0.6000    0.06360   0.6000    -0.02760
    0.7000    0.05180   0.7000    -0.02140
    0.8000    0.03750   0.8000    -0.01500
    0.9000    0.02080   0.9000    -0.0082
    0.9500    0.01140   0.9500    -0.00480
    1.0000    0.001300  1.0000    -0.00130
```

CFC1 OUT

NACA 2112 TEST CASE
BLADE COORDINATES AND DERIVATIVES

X	Y	XP	YP	XS	YS
0.000000	0.000000	0.000648	-0.010414	0.235015	0.014942
0.001282	-0.002217	0.026121	-0.043702	0.276264	-0.496534
0.005303	-0.008624	0.057745	-0.082912	0.430194	-0.249983
0.013673	-0.017221	0.120674	-0.083128	0.735209	0.211713
0.024658	-0.021265	0.187397	-0.063782	0.525545	0.116041
0.049711	-0.030096	0.234261	-0.053668	0.685044	0.128244
0.075053	-0.034702	0.290573	-0.040259	0.442194	0.131212
0.111659	-0.038076	0.321713	-0.029276	0.397349	0.058348
0.138640	-0.040554	0.358841	-0.020400	0.357301	0.097442
0.174536	-0.042055	0.391610	-0.010480	0.317038	0.102991
0.214874	-0.042665	0.420766	-0.001489	0.273203	0.063760
0.257431	-0.042436	0.445451	0.006151	0.229503	0.066877
0.302261	-0.041533	0.465524	0.011470	0.185396	0.039822
0.318751	-0.040216	0.481792	0.014725	0.139667	0.031304
0.398652	-0.038635	0.493212	0.018224	0.092939	0.040207
0.445335	-0.036054	0.500022	0.022077	0.045716	0.036184
0.494679	-0.034317	0.502176	0.025556	-0.001786	0.034443
0.543346	-0.031660	0.499705	0.026235	-0.048412	0.019220
0.592637	-0.028819	0.492653	0.029301	-0.094617	0.002422
0.640136	-0.025940	0.481130	0.029112	-0.140733	-0.003892
0.686973	-0.02311[illegible]	0.465082	0.028521	-0.185970	-0.009136
0.731651	-0.020359	0.443642	0.027557	-0.230305	-0.009599
0.774131	-0.017637	0.419910	0.026674	-0.273176	-0.007862
0.813476	-0.015117	0.391081	0.025829	-0.313547	-0.012421
0.850799	-0.012061	0.358446	0.024015	-0.351017	-0.024874
0.884210	-0.010428	0.322239	0.021679	-0.385982	-0.015681
0.913964	-0.008374	0.282761	0.019758	-0.417769	-0.033767
0.939663	-0.006656	0.240339	0.014551	-0.445673	-0.070224
0.961072	-0.005513	0.195464	0.009026	-0.468999	-0.016303
0.977934	-0.003627	0.148412	0.010103	-0.487521	0.011543
0.990161	-0.003645	0.099937	0.004395	-0.501142	-0.015681
0.997532	-0.002835	0.050184	0.001227	-0.509265	0.309566
1.000071	0.000000	0.000134	0.026456	-0.507766	-0.222163
0.997579	0.000459	-0.049303	0.065883	-0.499983	0.083266
0.990339	0.001930	-0.098100	0.020181	-0.493356	0.088250
0.978317	0.004150	-0.146004	0.029856	-0.481845	0.095699
0.961715	0.007628	-0.192593	0.038770	-0.466858	0.084099
0.940565	0.012013	-0.237548	0.046210	-0.448117	0.067726
0.915131	0.016853	-0.280403	0.052186	-0.423974	0.055006
0.885913	0.022224	-0.320652	0.057029	-0.395503	0.052374
0.852277	0.028000	-0.357962	0.060382	-0.363766	0.025685
0.815138	0.034037	-0.391915	0.062441	-0.327206	0.015981
0.775117	0.040227	-0.422127	0.063437	-0.287953	0.092787
0.732644	0.046342	-0.448372	0.062902	-0.246221	-0.013007
0.687539	0.052530	-0.470385	0.060861	-0.201950	-0.029267
0.640165	0.058331	-0.487957	0.057052	-0.155715	-0.039366
0.591590	0.063662	-0.500902	0.051163	-0.107602	-0.068051
0.542270	0.068347	-0.509013	0.044249	-0.058439	-0.071294
0.492113	0.072312	-0.512663	0.037005	-0.005598	-0.079581
0.441944	0.075553	-0.510162	0.027989	0.043297	-0.104833
0.392169	0.077760	-0.503312	0.016593	0.095653	-0.127262
0.333237	0.078710	-0.491328	0.003030	0.148562	-0.147945
0.295462	0.078305	-0.474171	-0.012017	0.200704	-0.155231
0.250239	0.076396	-0.451975	-0.026938	0.251113	-0.147134
0.207215	0.073919	-0.424921	-0.030874	0.299649	-0.136644
0.167019	0.069352	-0.393240	-0.053817	0.345285	-0.127057

0.130149	0.052506	-0.357047	-0.066077	0.392910	[illegible]
0.097063	0.055430	-0.316291	-0.077664	0.432628	[illegible]
0.068135	0.047410	-0.272223	-0.085657	0.471681	[illegible]
0.043772	0.038739	-0.223695	-0.091778	0.494088	[illegible]
0.024221	0.029496	-0.172917	-0.095590	0.626313	[illegible]
0.010751	0.019460	-0.098123	-0.198021	0.763160	[illegible]
0.004133	0.009288	-0.045167	-0.091418	0.331950	[illegible]
0.000995	0.002365	-0.020425	-0.046756	0.210839	[illegible]
0.000326	-0.000652	0.012887	-0.020544	0.260046	[illegible]
0.002912	-0.004947	0.040366	-0.000166	0.364134	[illegible]
0.008713	-0.012928	0.083786	-0.089501	0.617127	[illegible]
0.020523	-0.021006	0.157808	-0.071521	0.698710	[illegible]
0.038830	-0.027312	0.210778	-0.059294	0.460144	[illegible]
0.061830	-0.032566	0.258139	-0.046911	0.474734	[illegible]
0.089353	-0.036527	0.301630	-0.033201	0.315669	[illegible]
0.120919	-0.039315	0.340646	-0.025276	0.375755	[illegible]
0.156113	-0.041456	0.375717	-0.015752	0.338125	[illegible]
0.194562	-0.042492	0.406826	-0.005742	0.234906	[illegible]
0.235853	-0.042643	0.433647	0.002504	0.251475	[illegible]
0.279569	-0.042049	0.456178	0.009196	0.297541	[illegible]
0.325282	-0.040932	0.474371	0.013199	0.152721	[illegible]
0.372563	-0.039485	0.488074	0.016349	0.116355	[illegible]
0.420965	-0.037694	0.497195	0.020224	0.069378	[illegible]
0.470036	-0.035528	0.501663	0.023809	0.021930	[illegible]
0.519318	-0.033022	0.501512	0.027122	-0.025209	[illegible]
0.568358	-0.030252	0.496762	0.029006	-0.071527	[illegible]
0.616709	-0.027380	0.487473	0.029240	-0.117676	[illegible]
0.663925	-0.024521	0.473660	0.026906	-0.163517	[illegible]
0.709566	-0.021722	0.455406	0.028035	-0.208223	[illegible]
0.753291	-0.019016	0.432802	0.027047	-0.252011	[illegible]
0.794409	-0.016357	0.405993	0.026307	-0.293727	[illegible]
0.832788	-0.013866	0.375223	0.025056	-0.332546	[illegible]
0.867969	-0.011511	0.340772	0.022705	-0.368917	[illegible]
0.899586	-0.009379	0.302891	0.021018	-0.402241	[illegible]
0.927334	-0.007453	0.261893	0.017640	-0.432302	[illegible]
0.950919	-0.006028	0.218159	0.011128	-0.457961	[illegible]
0.970095	-0.005109	0.172134	0.004224	-0.478845	[illegible]
0.984659	-0.004128	0.121291	0.010120	-0.495019	[illegible]
0.994456	-0.003219	0.075114	0.007856	-0.506375	[illegible]
0.999381	-0.002673	0.025143	0.033506	-0.510538	[illegible]
0.999396	0.000358	-0.024685	0.010213	-0.503391	[illegible]
0.994558	0.001121	-0.073776	0.015549	-0.497190	[illegible]
0.984931	0.003161	-0.122142	0.025084	-0.488134	[illegible]
0.970603	0.006029	-0.169482	0.034454	-0.474653	[illegible]
0.951702	0.009829	-0.215301	0.042690	-0.458114	[illegible]
0.928385	0.014360	-0.259272	0.049354	-0.436737	[illegible]
0.900871	0.019479	-0.300678	0.051757	-0.410100	[illegible]
0.869403	0.025071	-0.339695	0.058901	-0.380226	[illegible]
0.834274	0.030991	-0.375367	0.061543	-0.346714	[illegible]
0.795813	0.037120	-0.407500	0.063101	-0.307766	[illegible]
0.754308	0.043341	-0.435759	0.063364	-0.267507	[illegible]
0.710386	0.049511	-0.459922	0.062985	-0.224295	[illegible]
0.664224	0.055479	-0.479739	0.059198	-0.179568	[illegible]
0.616338	0.061071	-0.495019	0.054349	-0.131939	[illegible]
0.567182	0.066090	-0.505573	0.047738	-0.082605	[illegible]
0.517232	0.070433	-0.511166	0.040719	-0.041116	[illegible]
0.466380	0.074055	-0.511723	0.032827	0.019572	[illegible]
0.416915	0.076797	-0.507367	0.022557	0.065691	[illegible]
0.357523	0.078317	-0.497970	0.010066	0.122036	[illegible]
0.319309	0.078704	-0.483359	-0.004163	0.174622	[illegible]
0.272778	0.077529	-0.463092	-0.019579	0.226167	[illegible]
0.228425	0.074889	-0.439044	-0.034042	0.275064	[illegible]
0.186727	0.070879	-0.409638	-0.047363	0.322774	[illegible]
0.148133	0.065597	-0.375739	-0.059969	0.363541	[illegible]
0.113105	0.059113	-0.337182	-0.072177	0.415975	[illegible]
0.082066	0.051568	-0.294709	-0.081839	0.447784	[illegible]
0.055361	0.043158	-0.248404	-0.088314	0.497615	[illegible]
0.033399	0.034161	-0.199611	-0.094096	0.561655	[illegible]
0.016539	0.024639	-0.137798	-0.102333	0.785346	[illegible]
0.006803	0.014169	-0.065633	-0.105308	0.539536	[illegible]
0.002263	0.005292	-0.031479	-0.071158	0.241167	[illegible]
0.000242	0.000689	-0.010236	-0.022026	0.210683	[illegible]

CFC2 OUT

NACA 2412 TEST CASE
T= 6.28319 EL= 0.07448 N= 64
BLADE COORDINATES AT GIVEN STAGGER ANGLE

X	Y	XP	YP	RC
0.0000	0.0000	0.0007	-0.0104	0.0017
0.0013	-0.0022	0.0262	-0.0437	-0.1052
0.0053	-0.0086	0.0579	-0.0828	-0.0909
0.0137	-0.0172	0.1202	-0.0630	0.7733
0.0291	-0.0242	0.1875	-0.0635	19.0463
0.0498	-0.0300	0.2343	-0.0534	8.0065
0.0751	-0.0346	0.2806	-0.0399	5.7448
0.1047	-0.0379	0.3218	-0.0269	3.9839
0.1391	-0.0404	0.3587	-0.0204	3.2543
0.1750	-0.0419	0.3918	-0.0100	3.0682
0.2119	-0.0424	0.4208	-0.0009	3.1576
0.2575	-0.0421	0.4454	0.0067	3.4624
0.3023	-0.0411	0.4655	0.0121	3.7920
0.3468	-0.0398	0.4818	0.0154	3.9521
0.3907	-0.0381	0.4932	0.0189	3.7912
0.4355	-0.0361	0.5000	0.0227	3.8541
0.4847	-0.0337	0.5021	0.0262	4.0651
0.5339	-0.0310	0.4997	0.0289	4.4115
0.5827	-0.0280	0.4926	0.0299	4.6209
0.6355	-0.0251	0.4811	0.0298	4.5510
0.6870	-0.0222	0.4650	0.0291	4.4103
0.7317	-0.0194	0.4445	0.0281	4.3534
0.7742	-0.0167	0.4199	0.0272	4.2707
0.8110	-0.0141	0.3911	0.0263	4.0948
0.8506	-0.0116	0.3581	0.0245	3.8759
0.8813	-0.0093	0.3222	0.0221	3.5951
0.9136	-0.0072	0.2827	0.0201	3.3260
0.9397	-0.0054	0.2403	0.0149	3.2241
0.9611	-0.0043	0.1954	0.0093	3.2973
0.9780	-0.0034	0.1484	0.0103	3.4364
0.9902	-0.0024	0.0998	0.0055	3.4915
0.9975	-0.0016	0.0502	0.0013	3.3039
1.0000	0.0013	0.0001	0.0265	2.9154
0.9976	0.0018	-0.0493	0.0058	2.5531
0.9903	0.0033	-0.0991	0.0261	2.2642
0.9783	0.0057	-0.1460	0.0297	2.0445
0.9617	0.0091	-0.1926	0.0385	1.8248
0.9406	0.0132	-0.2376	0.0459	1.6420
0.9151	0.0180	-0.2805	0.0518	1.5018
0.8855	0.0234	-0.3207	0.0566	1.4034
0.8522	0.0291	-0.3580	0.0599	1.3317
0.8154	0.0351	-0.3920	0.0619	1.2646
0.7754	0.0412	-0.4222	0.0629	1.1911
0.7326	0.0474	-0.4485	0.0623	1.1664
0.6875	0.0534	-0.4705	0.0603	1.0120
0.6404	0.0592	-0.4880	0.0564	0.9122
0.5918	0.0647	-0.5010	0.0505	0.7995
0.5422	0.0691	-0.5091	0.0436	0.6771
0.4920	0.0730	-0.5121	0.0363	0.5469
0.4418	0.0761	-0.5102	0.0273	0.5321
0.3920	0.0783	-0.5033	0.0159	0.4921
0.3431	0.0792	-0.4913	0.0024	0.4032
0.2957	0.0787	-0.4742	-0.0126	0.2982
0.2502	0.0767	-0.4519	-0.0275	0.2415
0.2071	0.0733	-0.4219	-0.0314	0.1936
0.1669	0.0686	-0.3932	-0.0543	0.1035
0.1301	0.0627	-0.3570	-0.0665	0.0508
0.0970	0.0556	-0.3162	-0.0781	0.0356
0.0681	0.0475	-0.2721	-0.0854	0.0420
0.0437	0.0368	-0.2236	-0.0921	0.0750
0.0242	0.0295	-0.1728	-0.0968	0.1671
0.0107	0.0195	-0.0980	-0.1081	-0.1390
0.0041	0.0093	-0.0450	-0.0915	0.0303
0.0010	0.0024	-0.0204	-0.0468	0.0065
0.0003	-0.0007	0.0129	-0.0205	0.0295
0.0029	-0.0049	0.0405	-0.0661	-0.1472
0.0087	-0.0129	0.0839	-0.0894	0.1264
0.0206	-0.0210	0.1579	-0.0713	0.0466
0.0389	-0.0273	0.2109	-0.0590	0.0315
0.0619	-0.0325	0.2582	-0.0466	0.0343
0.0894	-0.0364	0.3017	-0.0338	0.0624
0.1210	-0.0393	0.3407	-0.0248	0.1401
0.1562	-0.0413	0.3757	-0.0153	0.2197
0.1946	-0.0422	0.4068	-0.0052	0.2476
0.2359	-0.0423	0.4336	0.0031	0.3098
0.2796	-0.0417	0.4562	0.0098	0.4170
0.3253	-0.0405	0.4744	0.0138	0.5600
0.3726	-0.0390	0.4881	0.0170	0.8416
0.4210	-0.0371	0.4972	0.0209	1.0666
0.4701	-0.0349	0.5017	0.0245	1.0931
0.5194	-0.0323	0.5015	0.0276	1.1551
0.5684	-0.0295	0.4967	0.0297	1.3787
0.6167	-0.0266	0.4874	0.0299	1.7374
0.6640	-0.0237	0.4736	0.0295	2.0496
0.7096	-0.0208	0.4554	0.0286	2.4441
0.7532	-0.0180	0.4328	0.0277	3.0207
0.7944	-0.0154	0.4060	0.0268	4.1728
0.8328	-0.0128	0.3752	0.0255	6.1602
0.8680	-0.0104	0.3407	0.0231	4.1446
0.8996	-0.0082	0.3029	0.0214	8.5974
0.9273	-0.0062	0.2619	0.0180	7.5292
0.9509	-0.0048	0.2181	0.0114	6.0260
0.9701	-0.0038	0.1721	0.0094	6.7263
0.9847	-0.0028	0.1243	0.0103	7.3395
0.9945	-0.0019	0.0751	0.0080	7.3188
0.9994	-0.0014	0.0251	0.0335	7.3508
0.9994	0.0017	-0.0247	0.0102	8.4930
0.9946	0.0024	-0.0738	0.0155	11.4252
0.9849	0.0044	-0.1222	0.0249	17.5221
0.9706	0.0073	-0.1695	0.0312	30.3188
0.9517	0.0111	-0.2154	0.0424	46.9696
0.9284	0.0156	-0.2593	0.0490	50.2459
0.9008	0.0206	-0.3009	0.0544	61.4383
0.8694	0.0262	-0.3398	0.0585	95.8607
0.8342	0.0321	-0.3755	0.0611	74.7746
0.7958	0.0382	-0.4076	0.0626	42.5377
0.7543	0.0443	-0.4358	0.0628	27.5939
0.7103	0.0504	-0.4600	0.0615	18.6903
0.6642	0.0563	-0.4798	0.0580	15.0454
0.6163	0.0619	-0.4951	0.0537	16.5752
0.5671	0.0668	-0.5056	0.0471	38.4168
0.5171	0.0711	-0.5112	0.0401	-95.3991
0.4669	0.0747	-0.5118	0.0322	-375.1699
0.4168	0.0773	-0.5074	0.0219	10.1636
0.3674	0.0789	-0.4980	0.0094	8.8606
0.3192	0.0791	-0.4834	-0.0050	-17.5239
0.2727	0.0779	-0.4637	-0.0202	-2.7830
0.2283	0.0752	-0.4390	-0.0346	-1.3432
0.1866	0.0711	-0.4096	-0.0480	-1.2779
0.1461	0.0658	-0.3757	-0.0605	11.3959
0.1130	0.0593	-0.3371	-0.0726	0.7087
0.0829	0.0516	-0.2946	-0.0822	0.4959
0.0553	0.0432	-0.2483	-0.0866	0.4725
0.0334	0.0342	-0.1995	-0.0944	0.3208
0.0165	0.0247	-0.1377	-0.1025	-0.0587
0.0069	0.0142	-0.0655	-0.1054	0.0078
0.0023	0.0053	-0.0314	-0.0712	0.0026
0.0002	0.0007	-0.0102	-0.0220	0.0014

CFC2 OUT

NACA 2412 TEST CASE
T= 6.28319 EL= 0.07448 N= 64

INLET CONDITIONS
B1= 21.35 W1/W2= 1.00897
OUTLET CONDITIONS
B2= 7.21 W2/W1= 0.94719

NON DIMENSIONAL CIRCULATION = -1.30829

THE 2-D LIFT COEFFICIENT = 2.61659

THE 2-D PITCHING MOMENT COEFFICIENT =-0.0003540

VELOCITY DISTRIBUTION

SUCTION SIDE X/C	S/C	WL/W1	PRESSURE SIDE X/C	S/C	WL/W1
0.000	0.000	-11.113	0.000	0.000	-11.113
0.001	0.003	-4.398	0.001	0.003	-3.942
0.004	0.010	-4.024	0.005	0.010	-3.071
0.011	0.022	-4.258	0.014	0.022	-2.134
0.024	0.039	-3.417	0.029	0.039	-1.028
0.044	0.061	-2.669	0.050	0.061	-0.529
0.068	0.087	-2.580	0.075	0.086	-0.213
0.097	0.117	-2.376	0.105	0.116	0.002
0.130	0.151	-2.217	0.138	0.150	0.153
0.167	0.188	-2.079	0.175	0.187	0.268
0.207	0.228	-1.968	0.215	0.227	0.356
0.250	0.272	-1.872	0.257	0.269	0.425
0.296	0.317	-1.787	0.302	0.314	0.480
0.343	0.365	-1.706	0.349	0.360	0.526
0.392	0.413	-1.628	0.397	0.408	0.568
0.442	0.463	-1.556	0.445	0.457	0.603
0.492	0.514	-1.492	0.495	0.507	0.633
0.542	0.564	-1.436	0.544	0.556	0.658
0.592	0.614	-1.387	0.593	0.605	0.680
0.640	0.663	-1.337	0.640	0.653	0.700
0.688	0.710	-1.290	0.687	0.699	0.719
0.733	0.756	-1.246	0.732	0.744	0.737
0.775	0.799	-1.204	0.774	0.786	0.754
0.815	0.839	-1.165	0.814	0.826	0.769
0.852	0.877	-1.128	0.851	0.863	0.782
0.886	0.910	-1.091	0.884	0.897	0.796
0.915	0.940	-1.056	0.914	0.927	0.806
0.941	0.966	-1.022	0.940	0.952	0.813
0.962	0.988	-0.993	0.961	0.974	0.847
0.978	1.005	-0.976	0.978	0.991	0.897
0.990	1.017	-0.983	0.990	1.003	0.932
0.998	1.024	-1.171	0.998	1.010	1.171

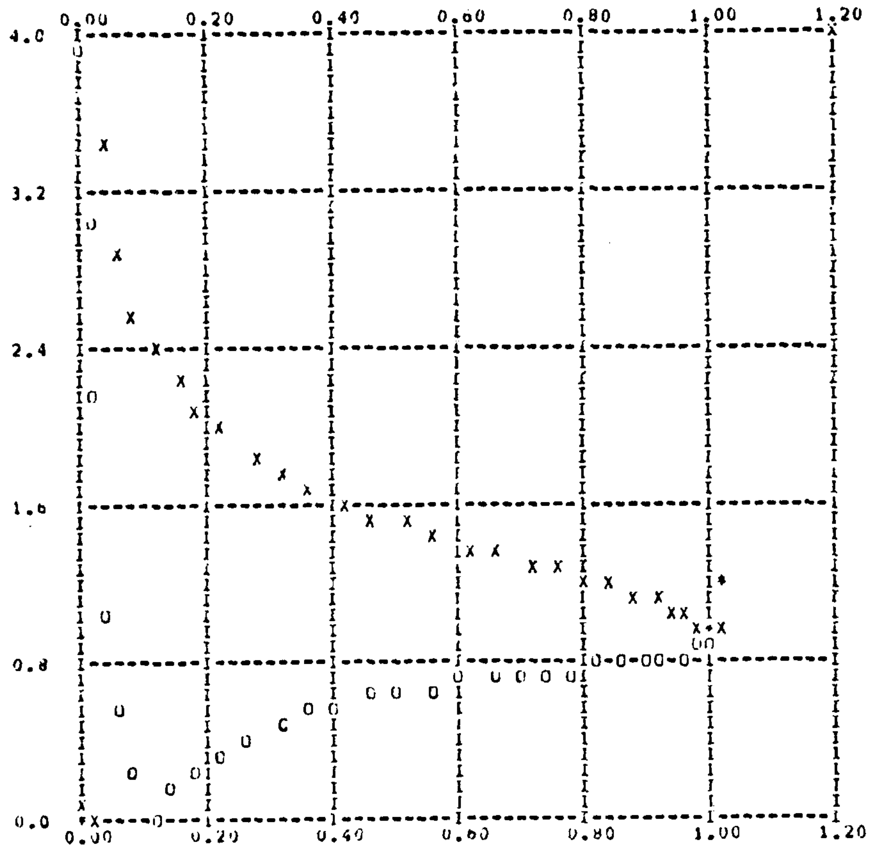
(P01-PSL)/Q1 DISTRIBUTION

SUCTION SIDE X/C	S/C	S LOC	ML	PRESSURE SIDE X/C	S/C	S LOC	ML
0.000	0.000	123.494	-11.113	0.000	0.000	123.494	-11.113
0.001	0.003	19.344	-4.398	0.001	0.003	15.537	-3.942
0.004	0.010	16.189	-4.024	0.005	0.010	9.379	-3.071
0.011	0.022	14.127	-4.258	0.014	0.022	4.553	-2.134
0.024	0.039	11.679	-3.417	0.029	0.039	1.057	-1.028
0.044	0.061	8.233	-2.669	0.050	0.061	0.269	-0.529
0.068	0.087	6.659	-2.580	0.075	0.086	0.015	-0.213
0.097	0.117	5.644	-2.376	0.105	0.116	0.000	0.002
0.130	0.151	4.917	-2.217	0.138	0.150	0.024	0.153
0.167	0.188	4.323	-2.079	0.175	0.187	0.072	0.268
0.207	0.228	3.872	-1.968	0.215	0.227	0.127	0.356
0.250	0.272	3.506	-1.872	0.257	0.269	0.181	0.425
0.296	0.317	3.194	-1.787	0.302	0.314	0.239	0.480
0.343	0.365	2.909	-1.706	0.349	0.360	0.277	0.526
0.392	0.413	2.650	-1.628	0.397	0.408	0.322	0.568
0.442	0.463	2.422	-1.556	0.445	0.457	0.363	0.603
0.492	0.514	2.225	-1.492	0.495	0.507	0.401	0.633
0.542	0.564	2.063	-1.436	0.544	0.556	0.433	0.658
0.592	0.614	1.923	-1.367	0.593	0.605	0.452	0.680
0.640	0.663	1.788	-1.337	0.640	0.653	0.471	0.700
0.688	0.710	1.664	-1.290	0.687	0.699	0.517	0.719
0.733	0.756	1.552	-1.246	0.732	0.744	0.544	0.737
0.775	0.799	1.450	-1.204	0.774	0.786	0.569	0.754
0.815	0.839	1.357	-1.165	0.814	0.826	0.592	0.769
0.852	0.877	1.272	-1.128	0.851	0.863	0.611	0.782
0.886	0.910	1.190	-1.091	0.884	0.897	0.634	0.796
0.915	0.940	1.115	-1.056	0.914	0.927	0.650	0.806
0.941	0.966	1.045	-1.022	0.940	0.952	0.690	0.813
0.962	0.988	0.985	-0.993	0.961	0.974	0.717	0.847
0.978	1.005	0.953	-0.976	0.978	0.991	0.805	0.897
0.990	1.017	0.967	-0.983	0.990	1.003	0.866	0.932
0.998	1.024	1.370	-1.171	0.998	1.010	1.370	1.171

CFC3 OUT

NACA 2412 TEST CASE
T= 0.28319 EL= 0.07448 N= 64

PLOT MACH NUMBER DISTRIBUTION



TAGNATION POINT IS AT 8 X= 0.10466
UCTION SIDE SEPARATION POINT IS AT 57 S/C= 0.15066

TAGNATION POINT IS AT 8 X= 0.10466
RESSURE SIDE SEPARATION POINT IS AT 33 S/C= 1.02761

OF ATTACK= 0.349

R OF ITERATIONS= 0

LATION= -0.69835

-D LIFT COEFFICIENT = 1.39671

-D FORM DRAG COEFFICIENT=0.64329350

KIN FRICTION DRAG COEFFICIENT=0.00000000

-D PITCHING MOMENT COEFFICIENT=-0.0303074

ITY DISTRIBUTION

UCTION SIDE			PRESSURE SIDE		
	S/C	WL/W1	X/C	S/C	WL/W1
00	0.000	-7.166	0.000	0.000	-7.166
01	0.003	-2.898	0.001	0.003	-2.467
04	0.010	-2.741	0.005	0.010	-1.615
11	0.022	-3.010	0.014	0.022	-1.070
24	0.039	-2.480	0.029	0.039	-0.336
44	0.061	-2.108	0.050	0.061	0.009
69	0.087	-1.901	0.075	0.086	0.214
97	0.117	-1.742	0.105	0.115	0.354
39	0.151	-1.570	0.138	0.150	0.456
67	0.186	-1.414	0.175	0.187	0.534
07	0.228	-1.321	0.215	0.227	0.593
50	0.272	-1.260	0.257	0.269	0.639
96	0.317	-1.229	0.302	0.314	0.677
43	0.365	-1.201	0.349	0.356	0.711
92	0.413	-1.268	0.397	0.405	0.743
47	0.463	-1.278	0.445	0.457	0.771
92	0.511	-1.263	0.495	0.507	0.796
42	0.554	-1.251	0.543	0.556	0.816
92	0.614	-1.239	0.593	0.605	0.834
40	0.663	-1.224	0.640	0.653	0.853
68	0.710	-1.209	0.687	0.699	0.870
33	0.756	-1.196	0.732	0.744	0.849
75	0.799	-1.184	0.774	0.786	0.907
15	0.839	-1.174	0.814	0.826	0.924
52	0.877	-1.166	0.851	0.863	0.940
86	0.910	-1.158	0.884	0.897	0.960
15	0.940	-1.152	0.914	0.927	0.975
41	0.966	-1.142	0.940	0.952	0.987
62	0.988	-1.152	0.961	0.974	1.035
78	1.005	-1.170	0.978	0.991	1.103

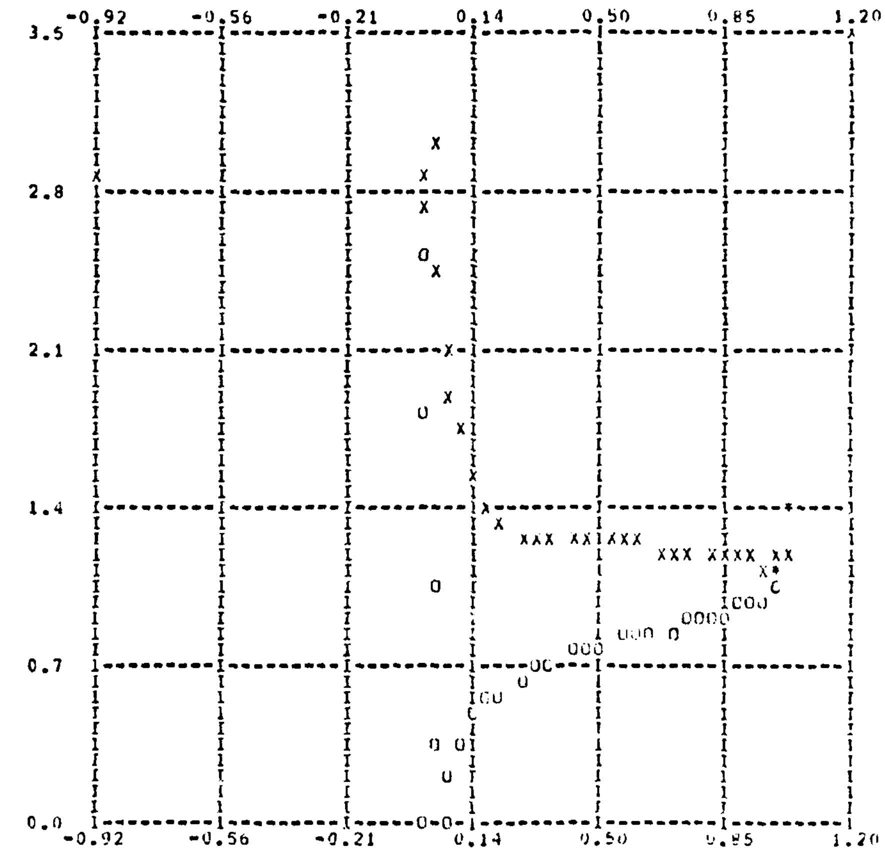
90	1.017	-1.221	0.990	1.003	1.144
98	1.024	-1.419	0.998	1.010	1.366

SL)/Q1 DISTRIBUTION

SUCTIONSIDE	S/C	S LOC	PRESSURESIDE X/C	S/C	S LOC
00	0.000	51.353	0.000	0.000	51.353
01	0.003	8.398	0.001	0.003	6.187
04	0.010	7.515	0.005	0.010	3.295
11	0.022	9.059	0.011	0.022	1.145
24	0.039	6.149	0.029	0.039	0.113
44	0.061	4.445	0.050	0.061	0.000
68	0.087	3.613	0.075	0.086	0.046
97	0.117	3.034	0.105	0.116	0.125
30	0.151	2.465	0.139	0.150	0.208
67	0.188	1.998	0.175	0.187	0.286
07	0.228	1.744	0.215	0.227	0.352
50	0.272	1.588	0.257	0.269	0.409
96	0.317	1.510	0.302	0.314	0.458
43	0.365	1.550	0.349	0.360	0.505
92	0.413	1.659	0.397	0.408	0.552
42	0.463	1.632	0.445	0.457	0.595
92	0.511	1.596	0.495	0.507	0.633
12	0.561	1.565	0.544	0.556	0.666
92	0.611	1.535	0.593	0.605	0.696
40	0.663	1.498	0.640	0.653	0.727
49	0.710	1.462	0.687	0.699	0.758
33	0.756	1.431	0.732	0.744	0.790
75	0.799	1.403	0.774	0.786	0.823
15	0.839	1.379	0.814	0.826	0.855
52	0.877	1.360	0.851	0.863	0.888
86	0.910	1.341	0.884	0.897	0.921
15	0.940	1.328	0.914	0.927	0.951
41	0.966	1.329	0.940	0.952	0.975
62	0.988	1.326	0.961	0.974	1.071
78	1.005	1.370	0.978	0.991	1.217
90	1.017	1.490	0.990	1.003	1.309
98	1.024	2.012	0.998	1.010	1.866

2412 TEST CASE
.28319 FL= 0.07448 N= 64

NON DIMENSIONAL VELOCITY DISTRIBUTION



THE MODEL

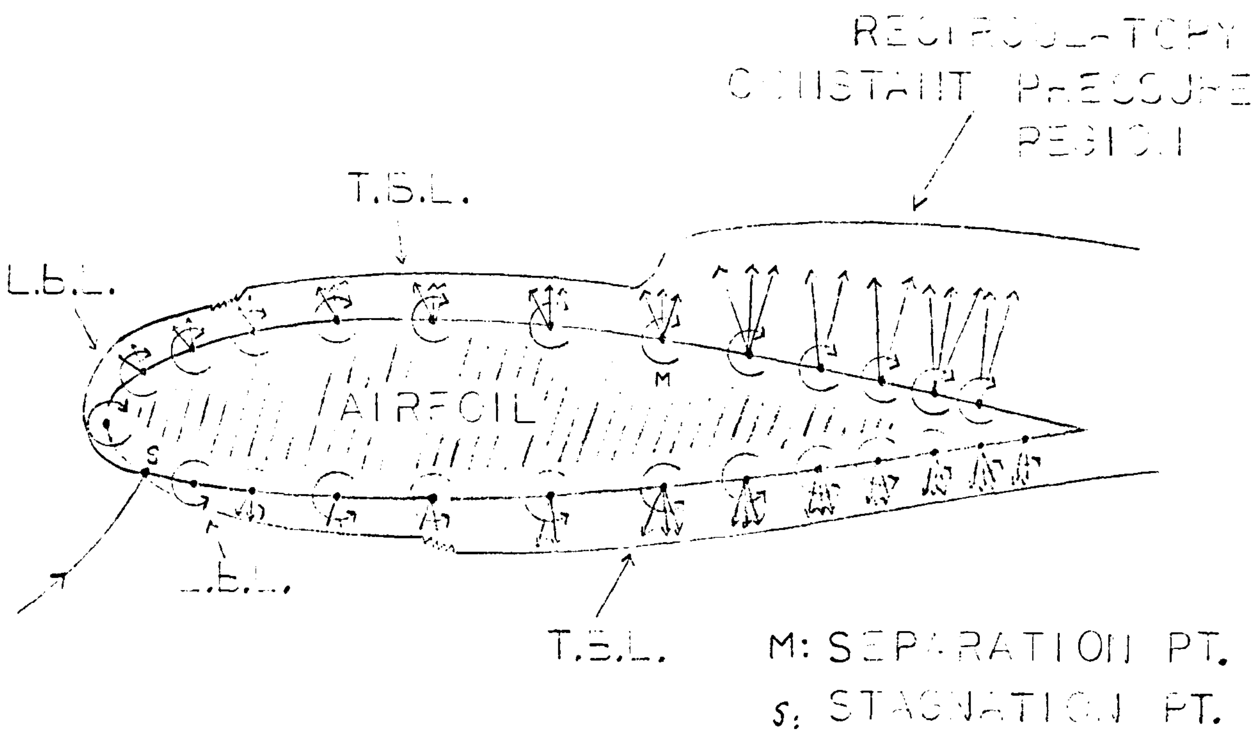


Fig. 1. The Model

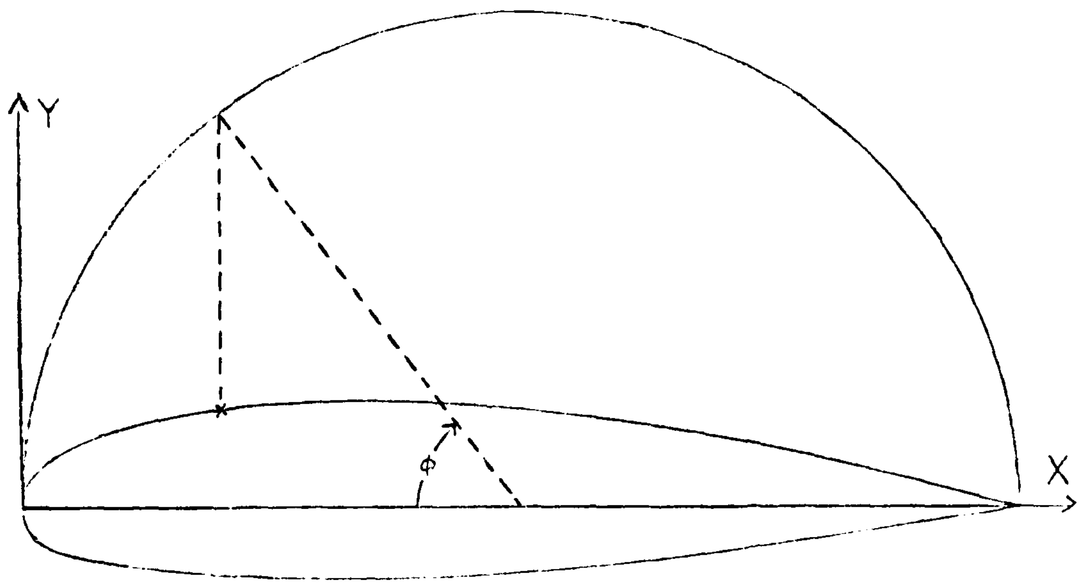


Fig. 2. The Airfoil Coordinate System

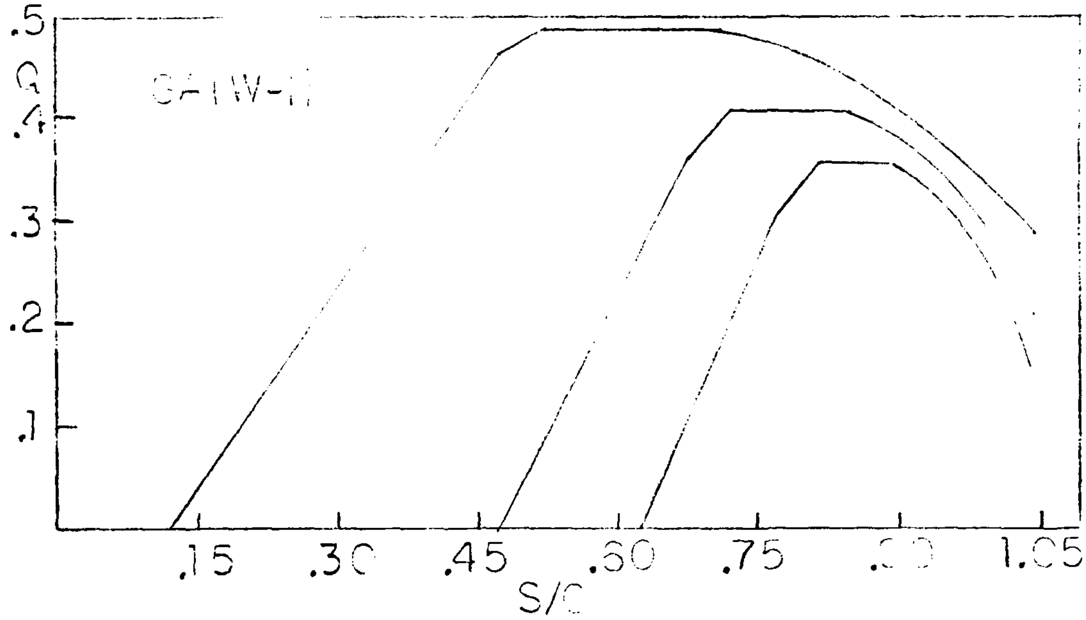
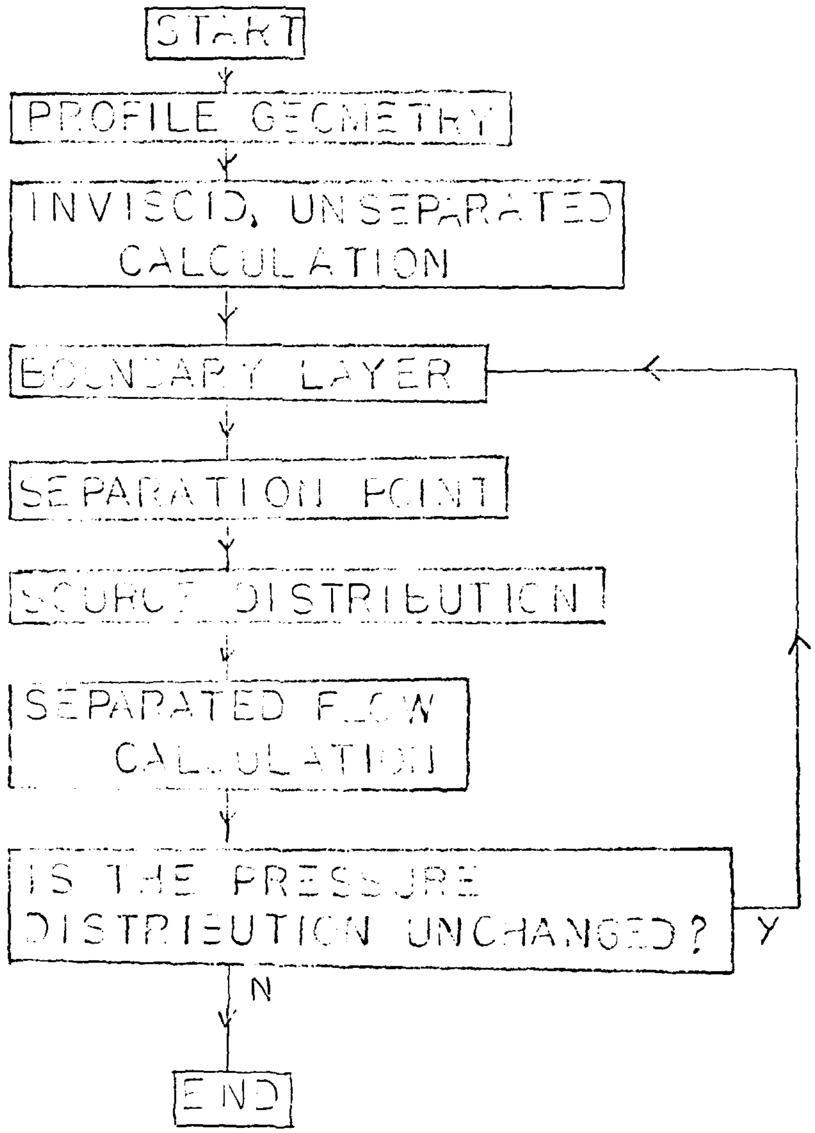


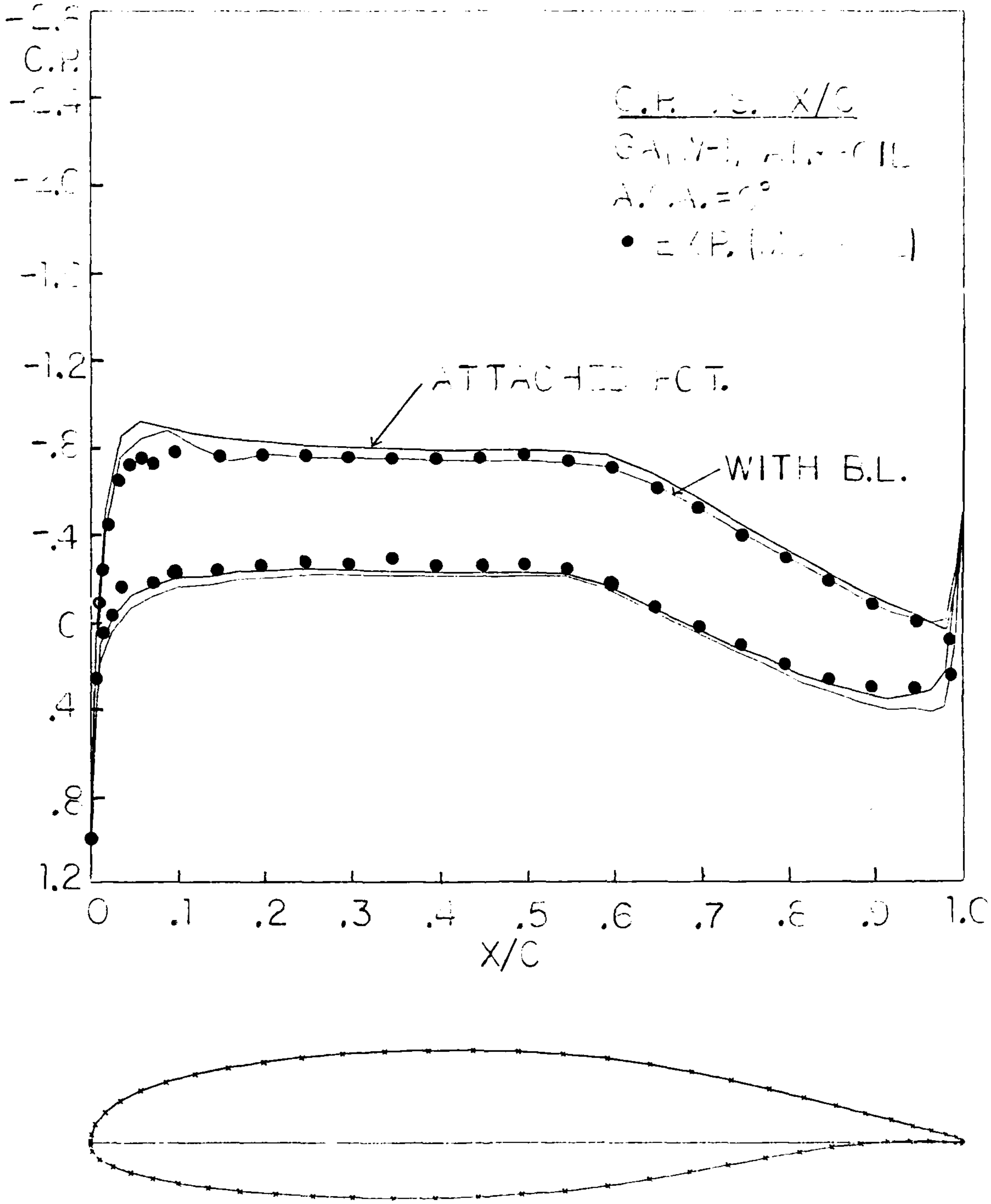
Fig. 3. Source Distributions for the GA(W-1)

THE COMPUTER PROGRAMS

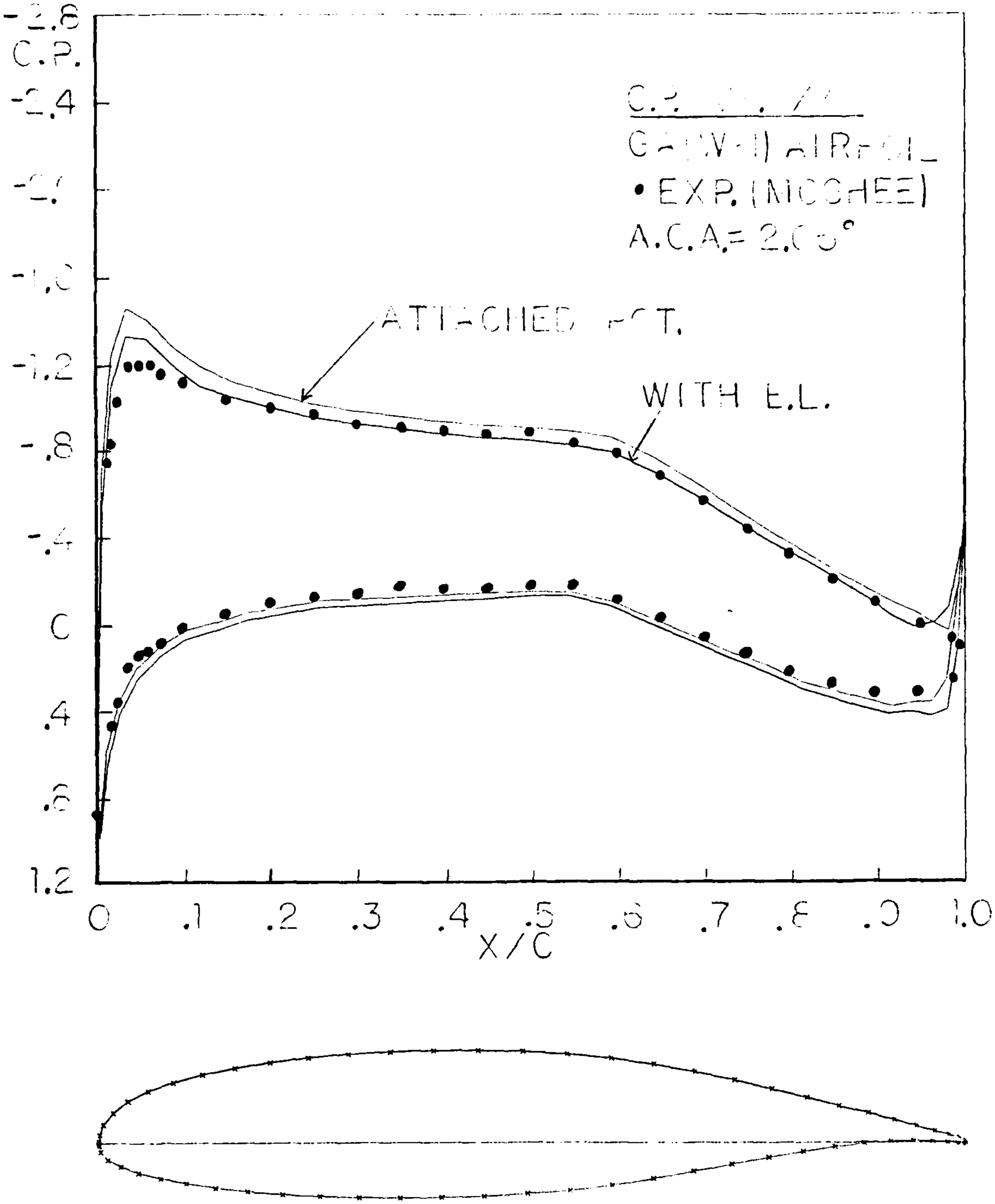
THE FOLLOWING FLOW CHART ILLUSTRATES THE COMPUTATIONAL PROCEDURE



Fig, 4. The Basic Computational Procedure



Fig, 5, Pressure Coefficient for GA(W-1)



Fig, 6, Pressure Coefficient for GA(W-1)

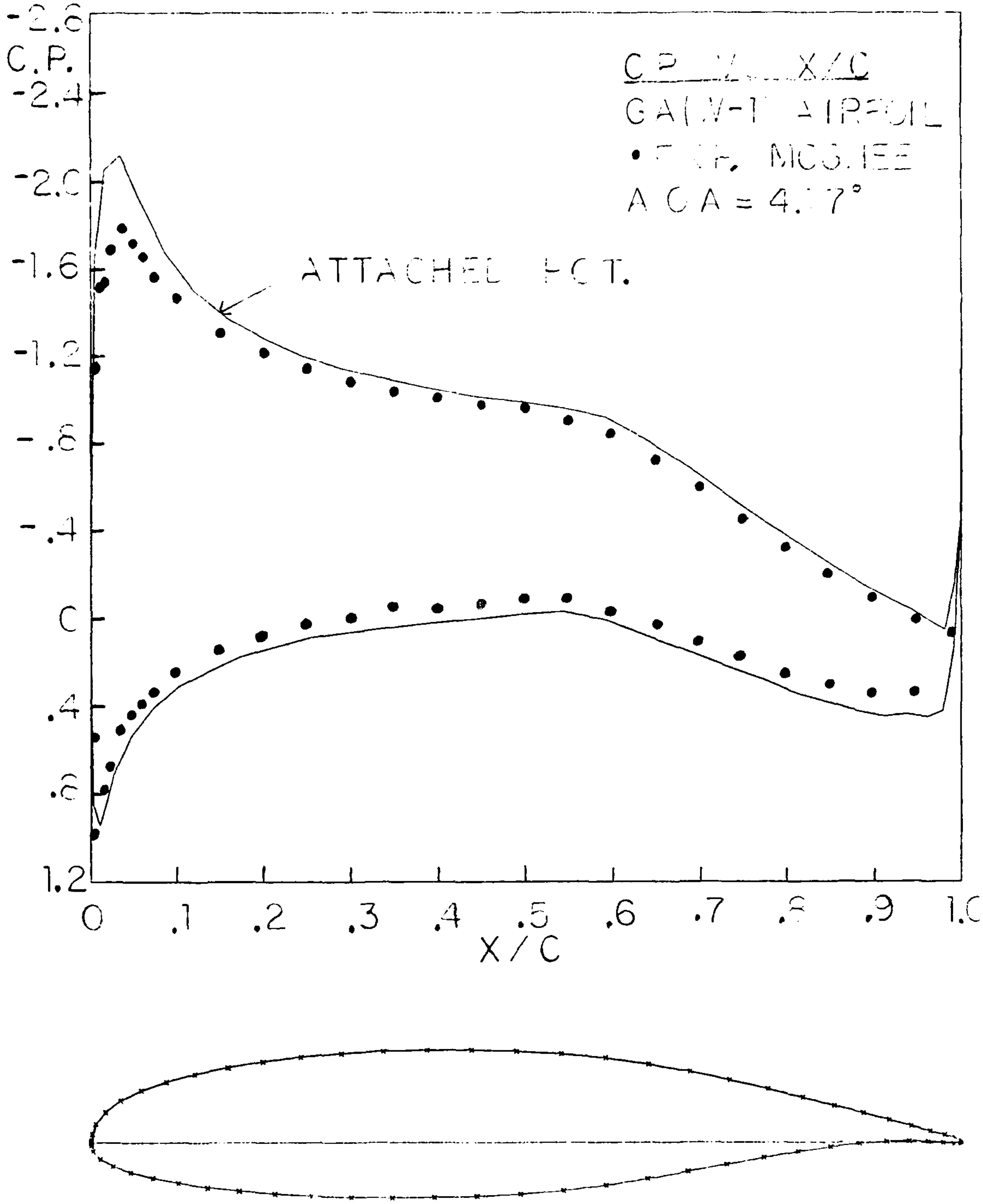


Fig. 7. Pressure Coefficient for GA(W-1)

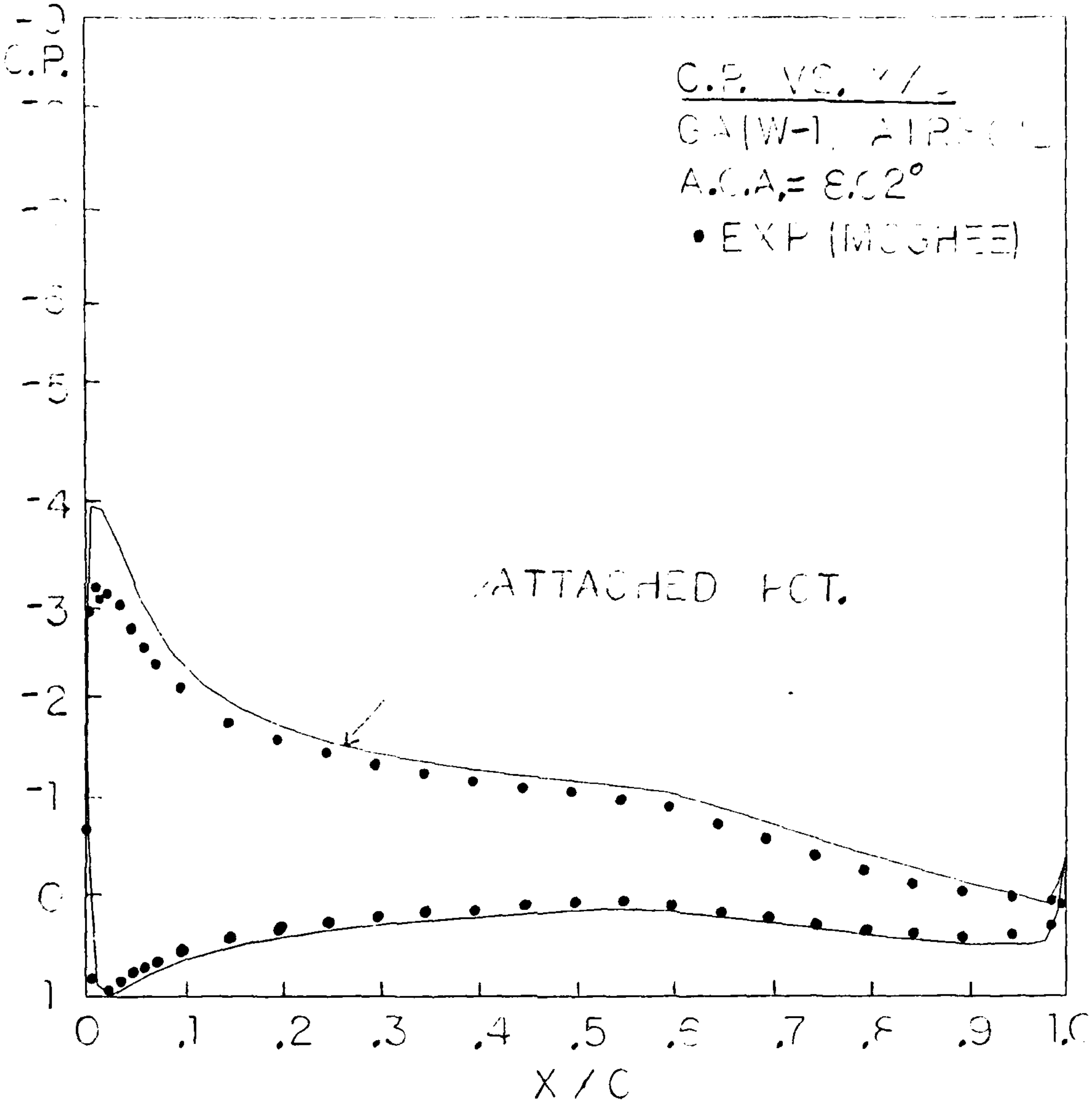


Fig. 8. Pressure Coefficient for GA(W-1)

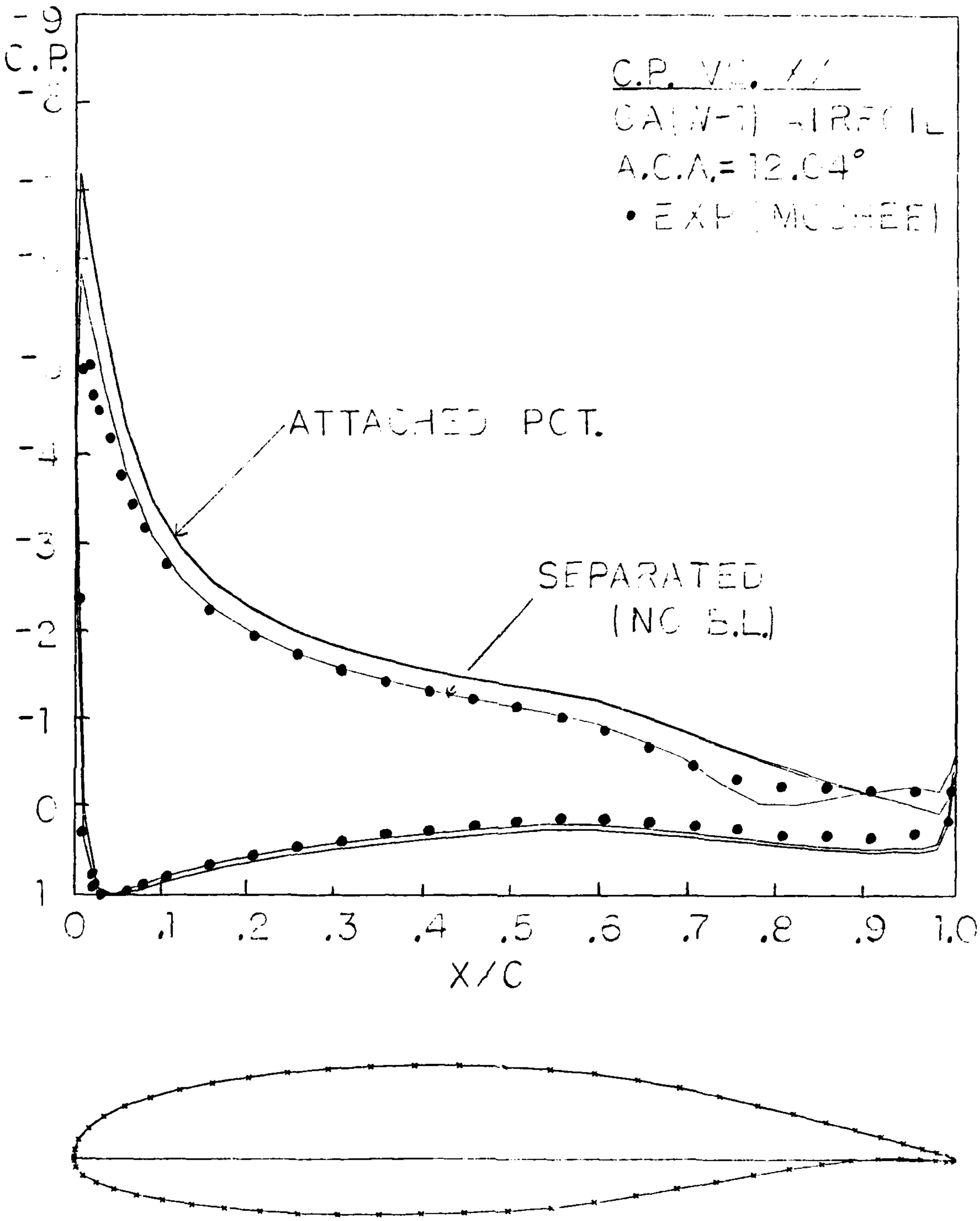


Fig. 9. Pressure Coefficient for GA(W-1)

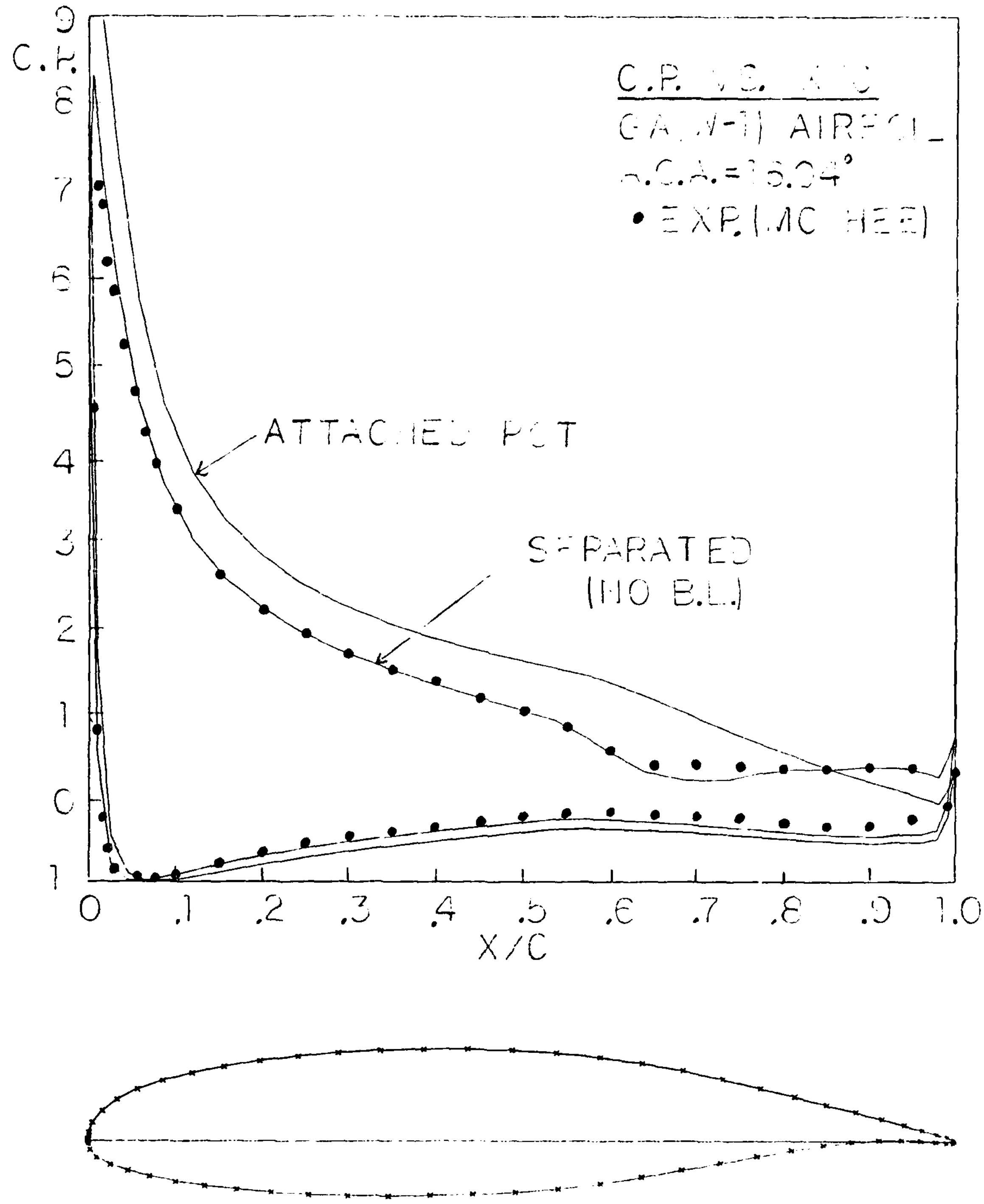


Fig. 10. Pressure coefficient for GA(W-1)

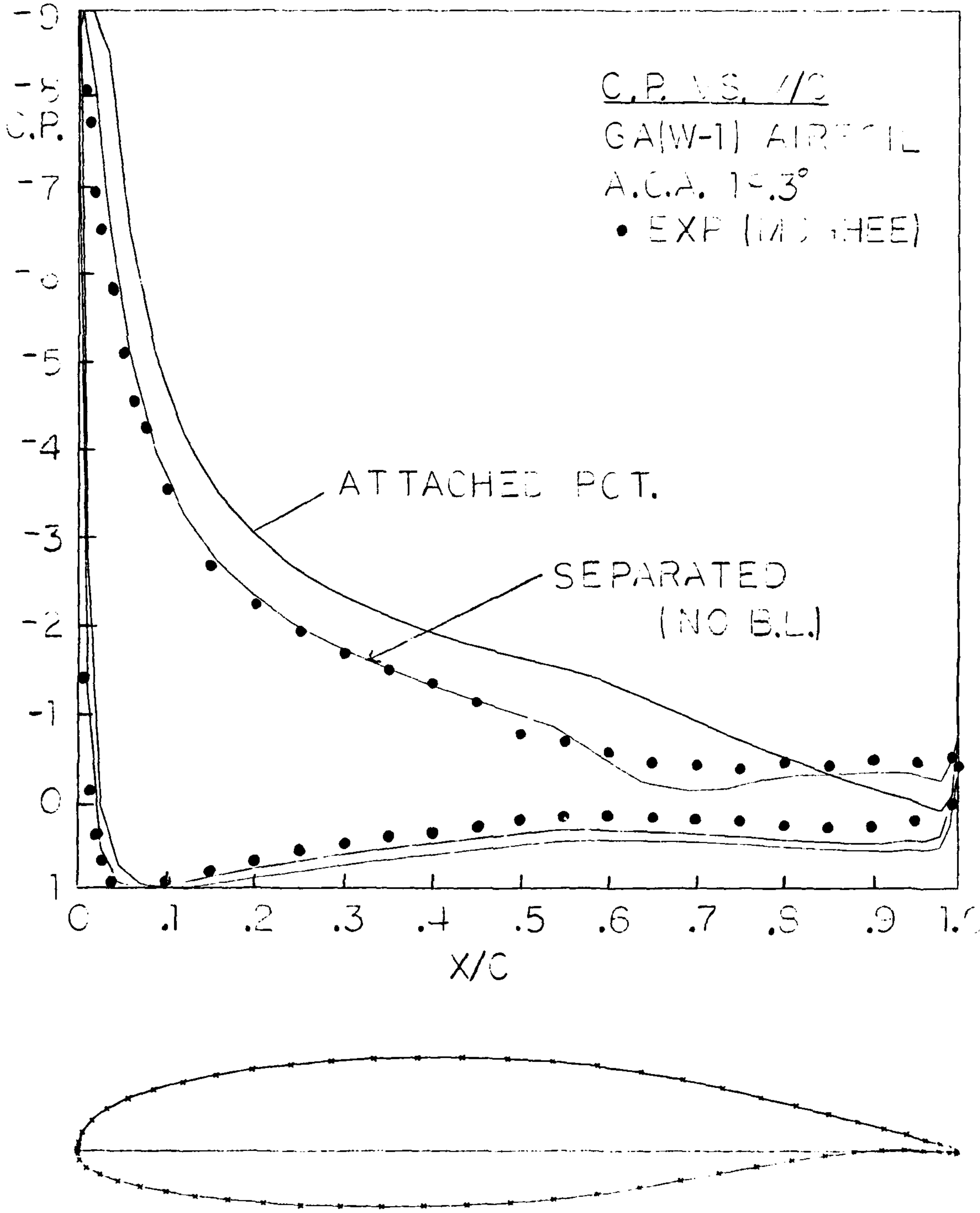


Fig. 11. Pressure Coefficient for GA(W-1)

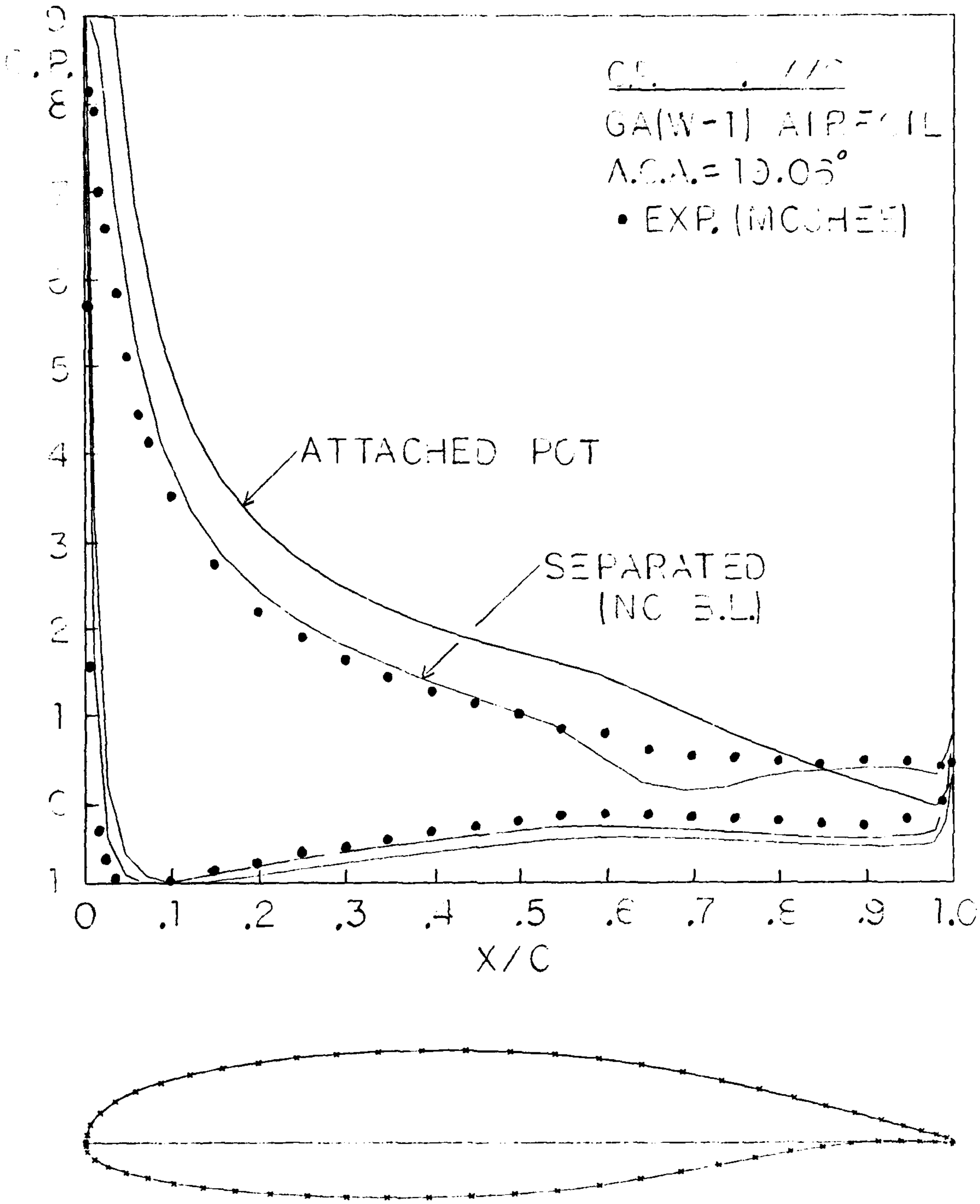


Fig. 12. Pressure Coefficient for GA(W-1)

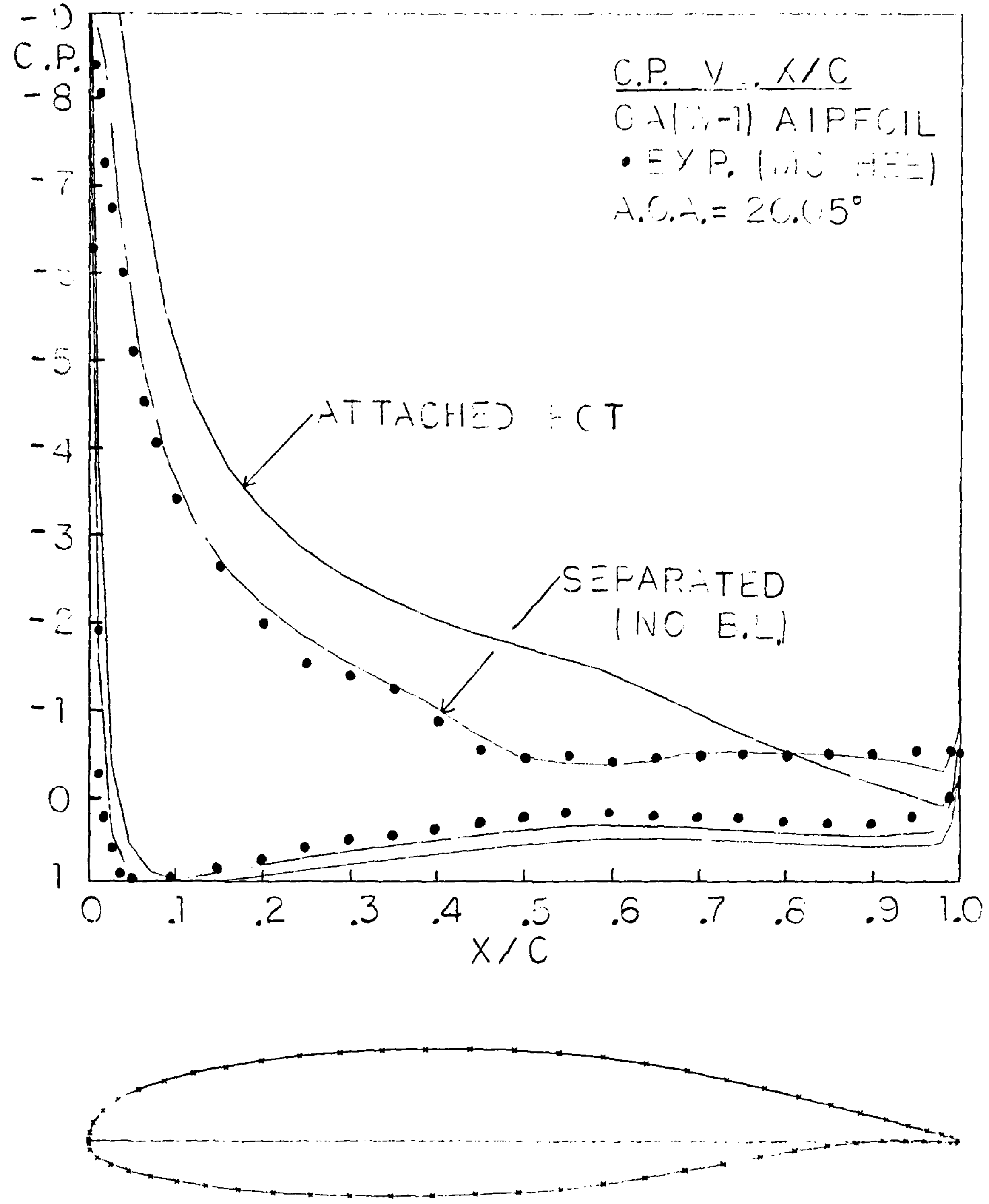
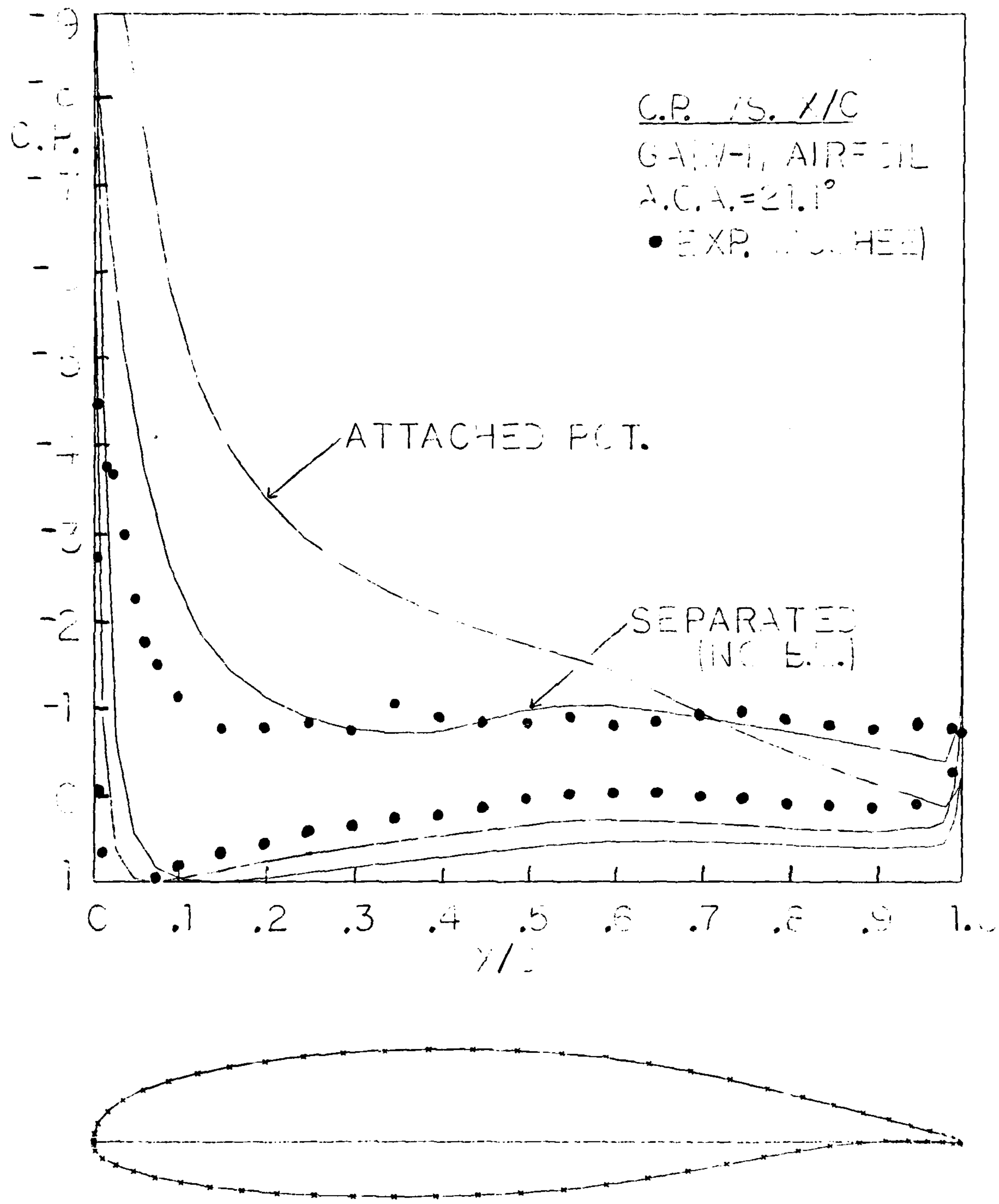


Fig. 13, Pressure Coefficient for GA(W-1)



Fig, 14. Pressure Coefficient for GA(W-1)

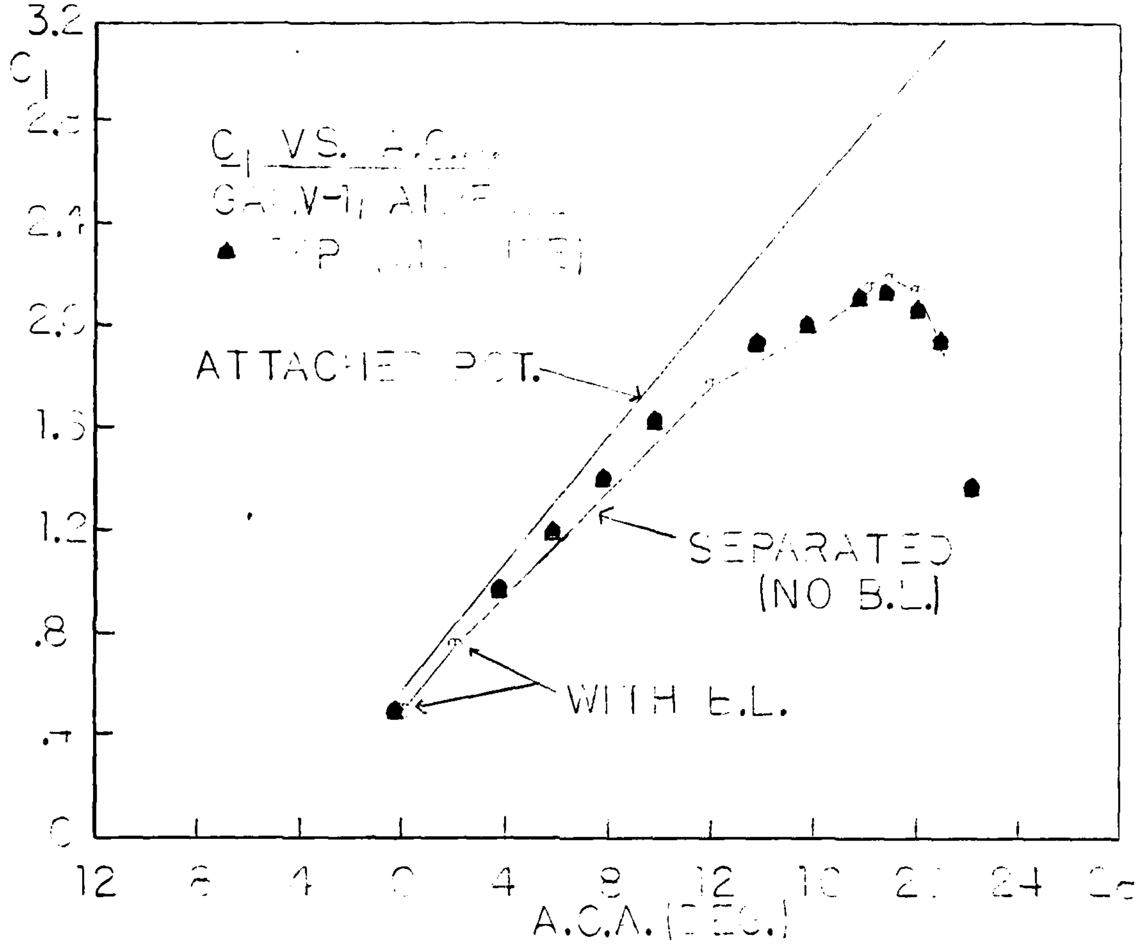


Fig. 15. Lift Coefficient for GA(W-1)

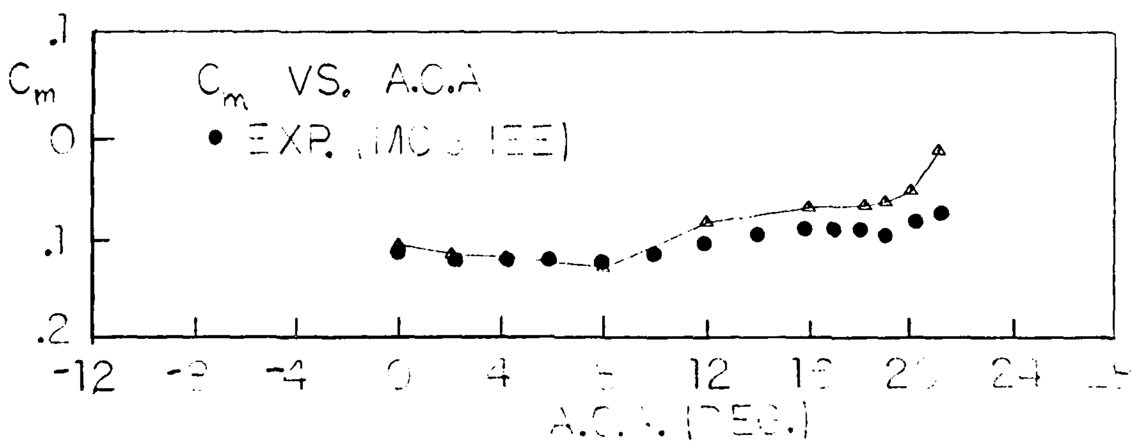


Fig. 16. Pitching Moment Coefficient for GA(W-1)

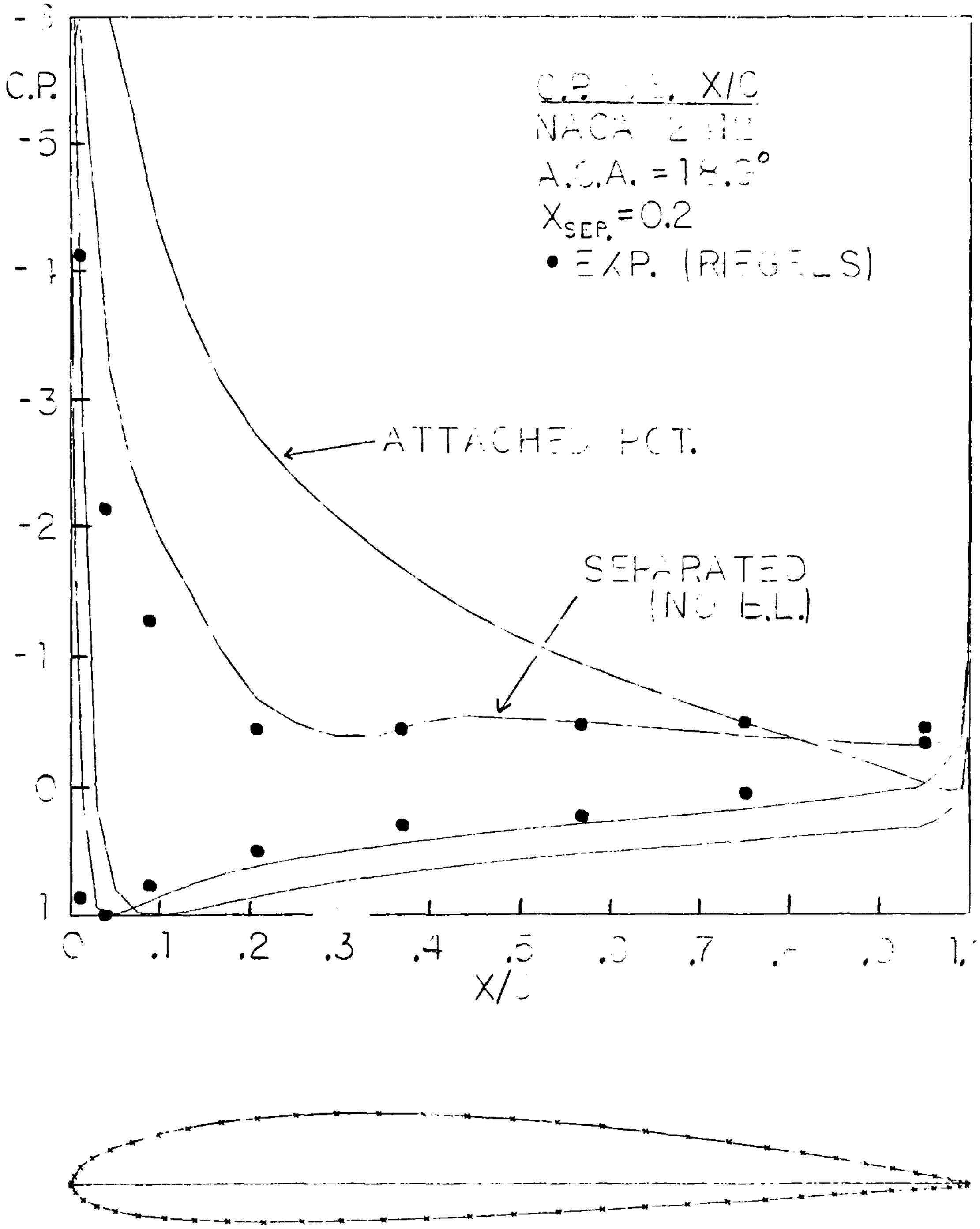


Fig. 17. Pressure Coefficient for NACA 2412

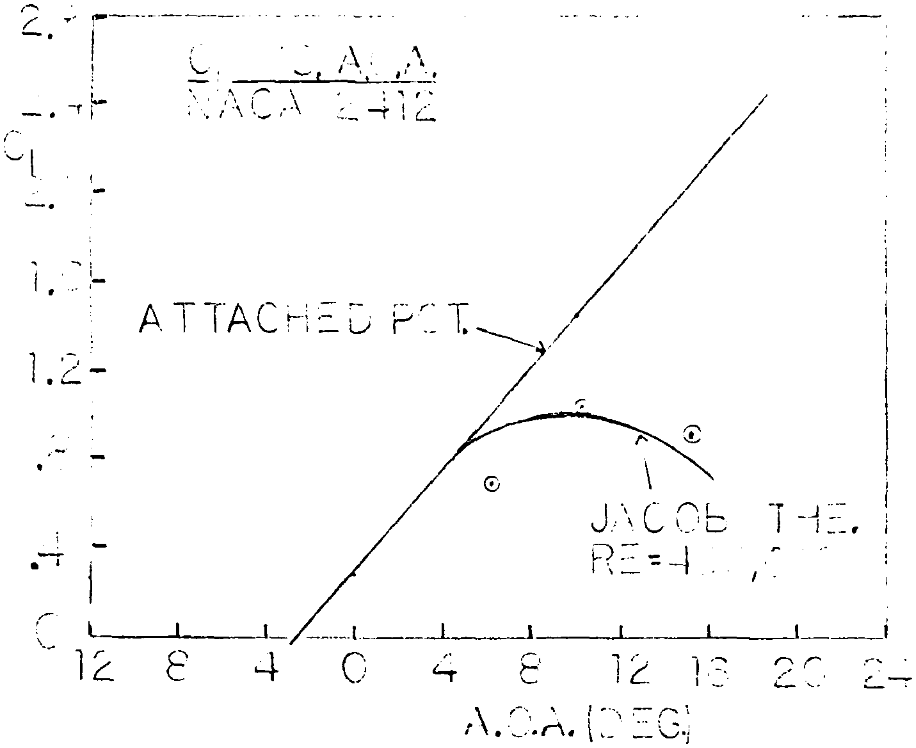


Fig. 18. Lift Coefficient for NACA 2412

END 5-83